6,665,297 B1

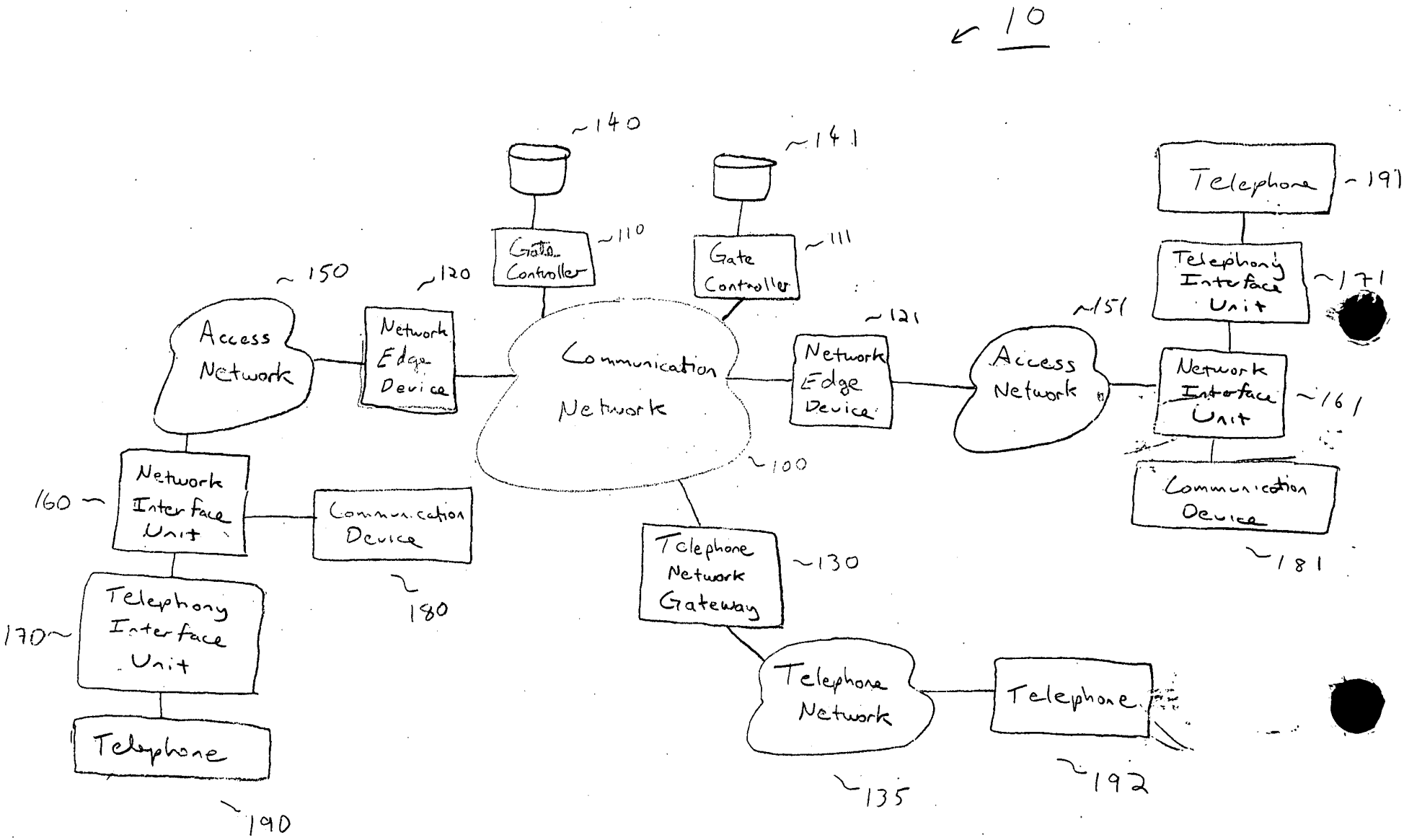

FIG. 1

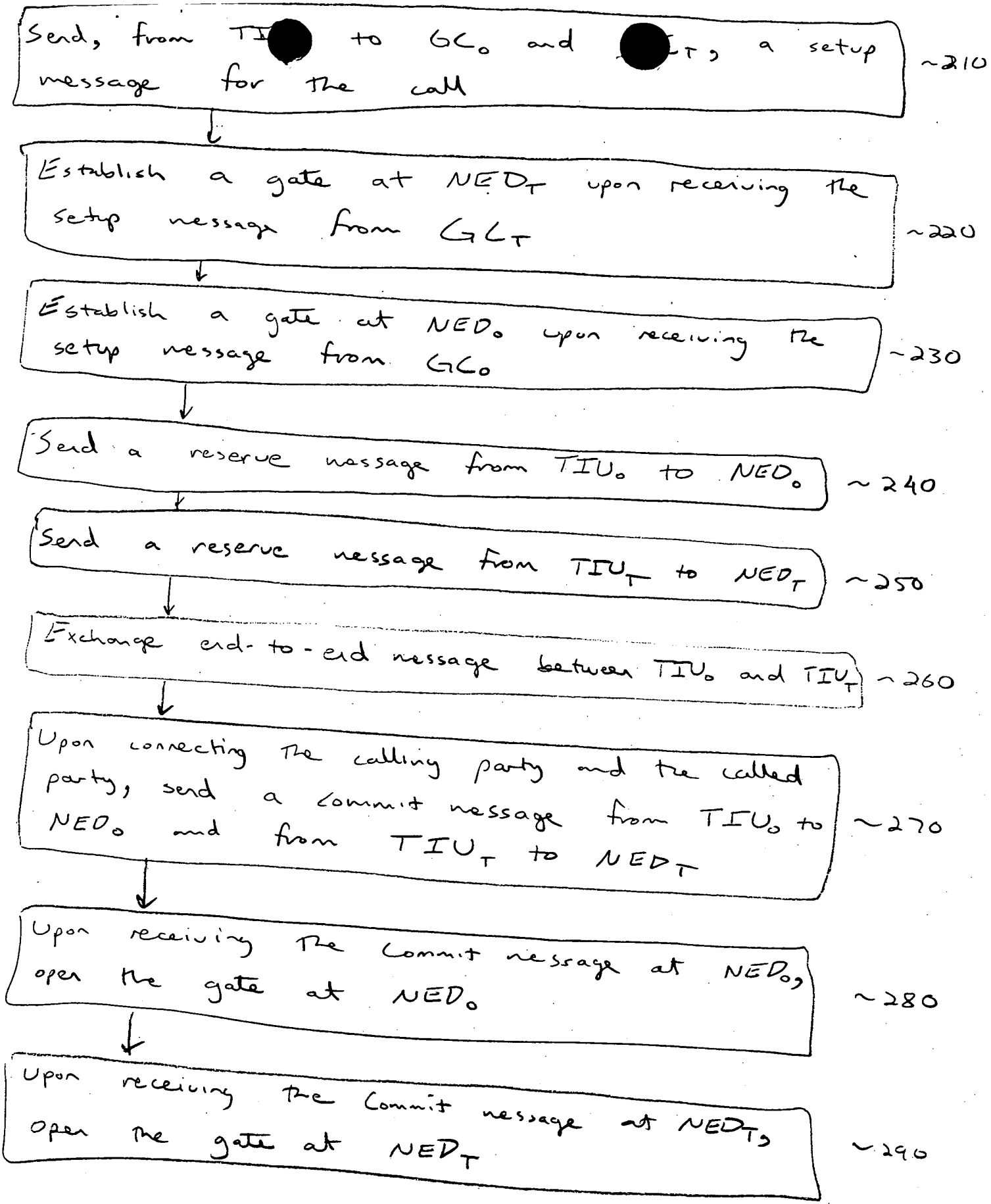
Send, from TI[redacted] to GC$_o$ and [redacted]$_T$, a setup message for the call
~210
Establish a gate at NED$_T$ upon receiving the setup message from GL$_T$
~220
Establish a gate at NED$_o$ upon receiving the setup message from GC$_o$
~230
Send a reserve message from TIU$_o$ to NED$_o$
~240
Send a reserve message from TIU$_T$ to NED$_T$
~250
Exchange end-to-end message between TIU$_o$ and TIU$_T$
~260
Upon connecting the calling party and the called party, send a commit message from TIU$_o$ to NED$_o$ and from TIU$_T$ to NED$_T$
~270
Upon receiving the commit message at NED$_o$, open the gate at NED$_o$
~280
Upon receiving the commit message at NED$_T$, open the gate at NED$_T$
~290

FIG. 2

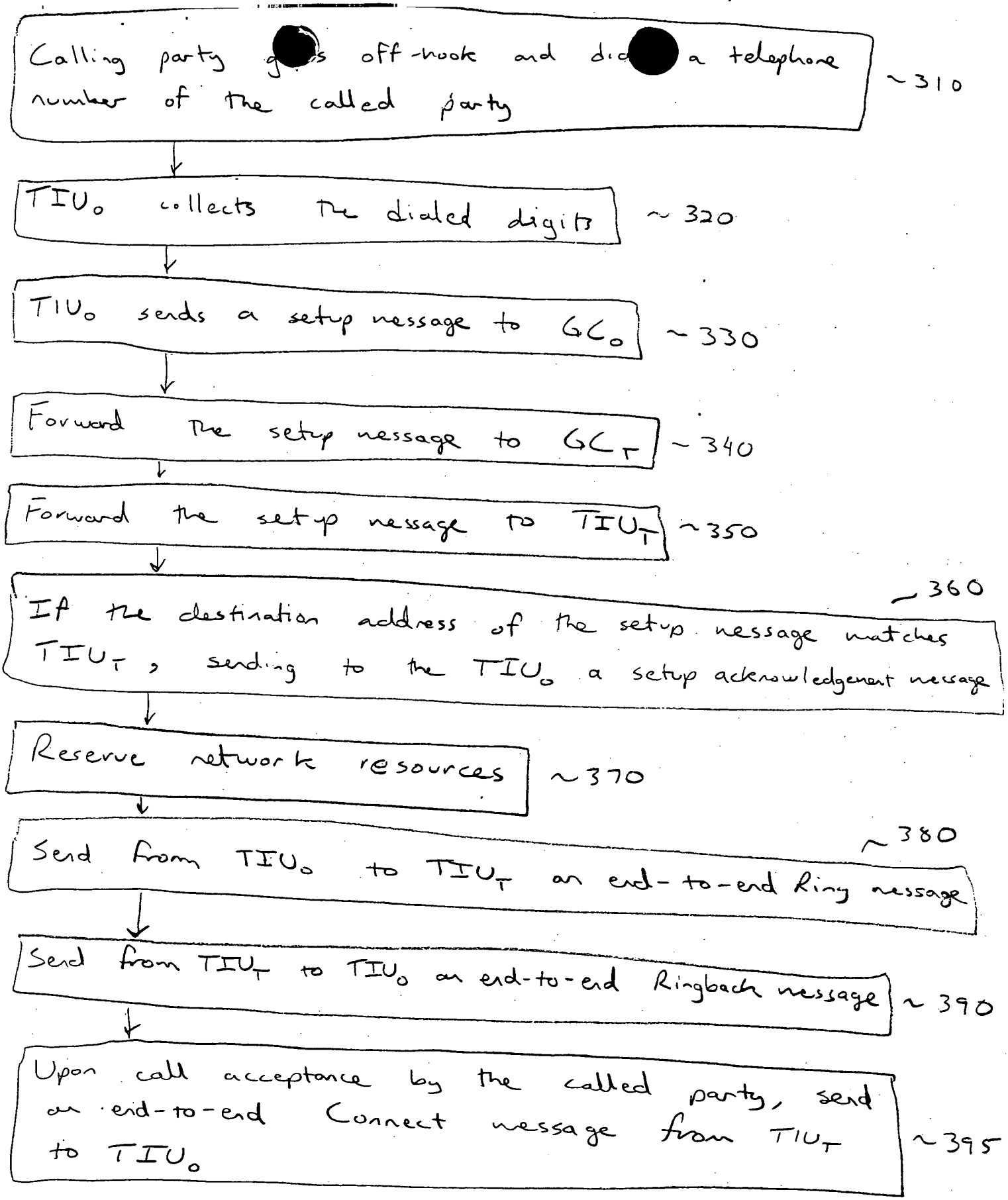
Calling party goes off-hook and dials a telephone number of the called party
~310
TIU_o collects the dialed digits
~320
TIU_o sends a setup message to GC_o
~330
Forward the setup message to GC_T
~340
Forward the setup message to TIU_T
~350
If the destination address of the setup message matches TIU_T, sending to the TIU_o a setup acknowledgement message
~360
Reserve network resources
~370
Send from TIU_o to TIU_T an end-to-end Ring message
~380
Send from TIU_T to TIU_o an end-to-end Ringback message
~390
Upon call acceptance by the called party, send an end-to-end Connect message from TIU_T to TIU_o
~395

FIG. 3

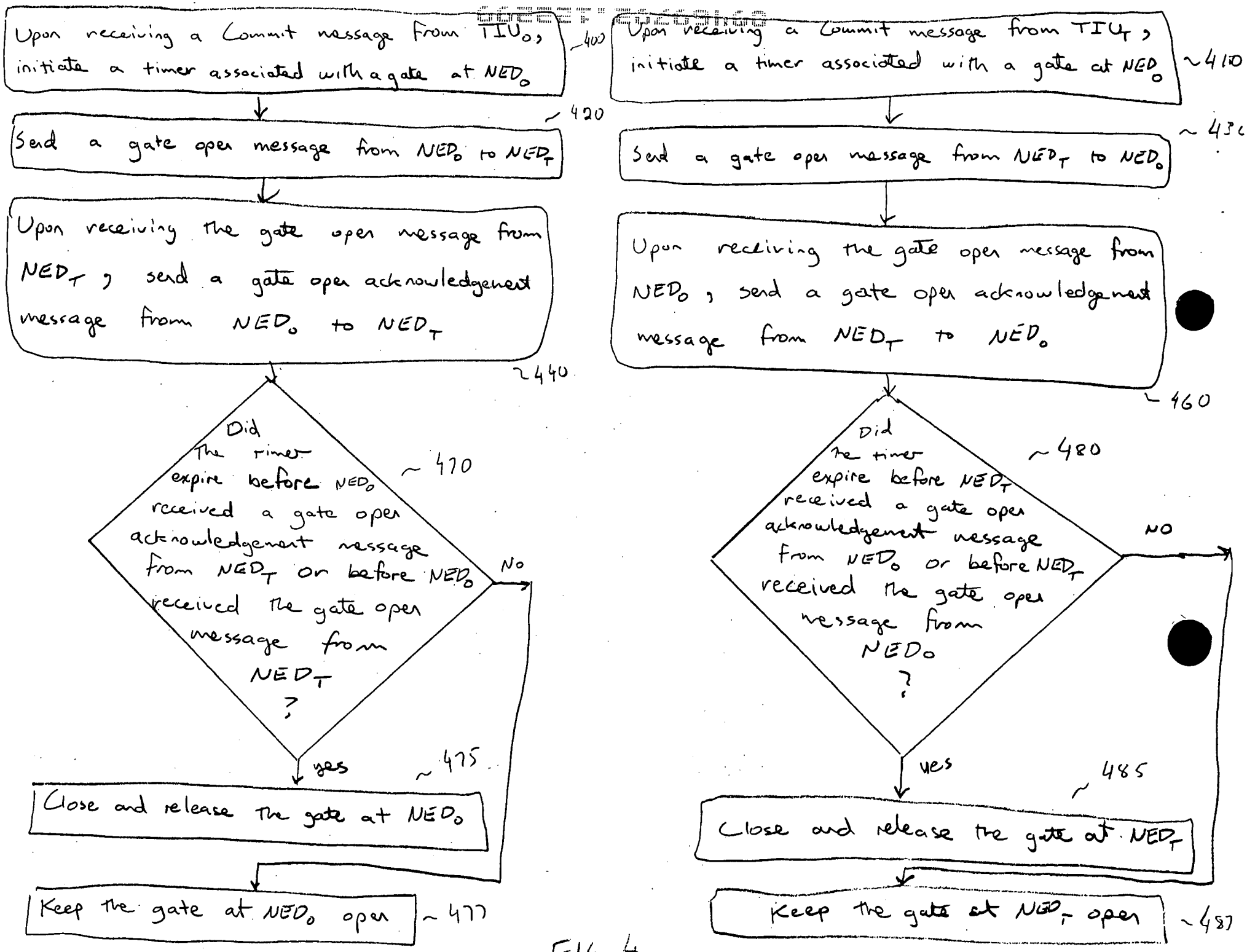
Upon receiving a Commit message from TIU0, initiate a timer associated with a gate at NED0
400
Upon receiving a Commit message from TIU4, initiate a timer associated with a gate at NED0
410
Send a gate open message from NED0 to NEDT
420
Send a gate open message from NEDT to NED0
430
Upon receiving the gate open message from NEDT, send a gate open acknowledgement message from NED0 to NEDT
440
Upon receiving the gate open message from NED0, send a gate open acknowledgement message from NEDT to NED0
460
Did the timer expire before NED0 received a gate open acknowledgement message from NEDT or before NED0 received the gate open message from NEDT ?
470
No
yes
475
Close and release the gate at NED0
Keep the gate at NED0 open
477
Did the timer expire before NEDT received a gate open acknowledgement message from NED0 or before NEDT received the gate open message from NED0 ?
480
NO
yes
485
Close and release the gate at NEDT
Keep the gate at NEDT open
487

FIG. 4

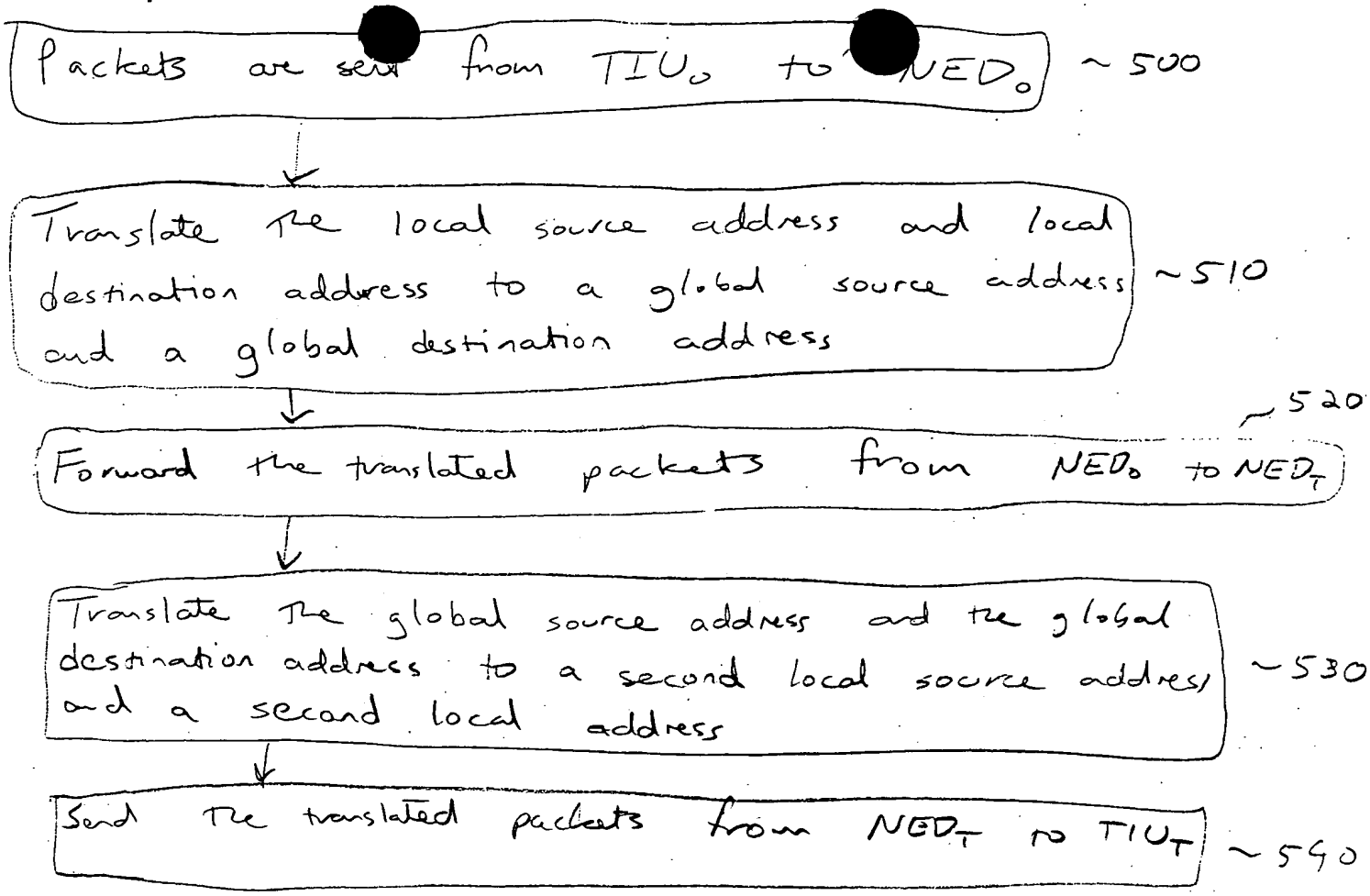
Packets are sent from TIU_o to NED_o
~500
Translate the local source address and local destination address to a global source address and a global destination address
~510
Forward the translated packets from NED_o to NED_T
~520
Translate the global source address and the global destination address to a second local source address and a second local address
~530
Send the translated packets from NED_T to TIU_T
~540

FIG. 5

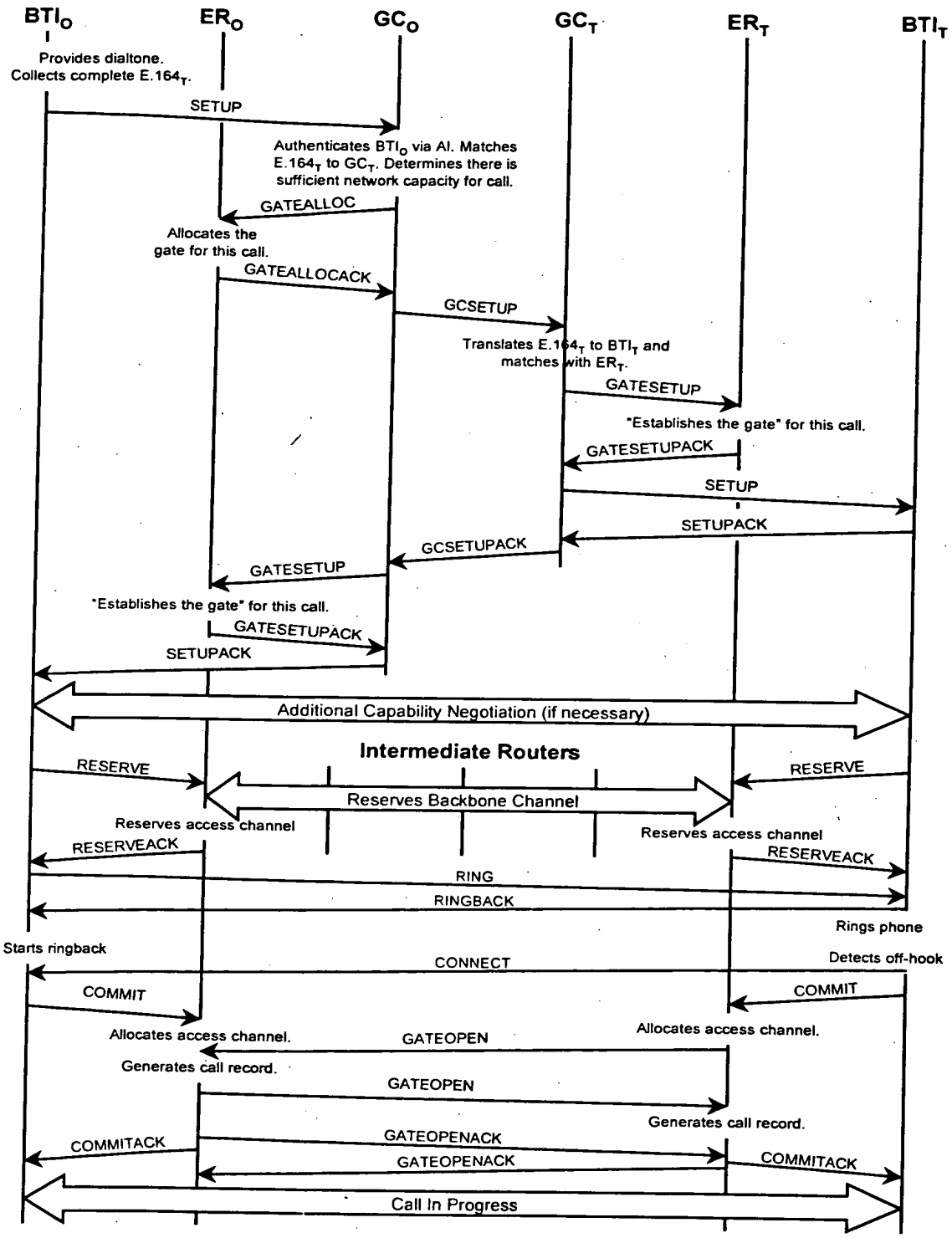
BTI_O
ER_O
GC_O
GC_T
ER_T
BTI_T
Provides dialtone.
Collects complete E.164_T.
SETUP
Authenticates BTI_O via AI. Matches E.164_T to GC_T. Determines there is sufficient network capacity for call.
GATEALLOC
Allocates the gate for this call.
GATEALLOCACK
GCSETUP
Translates E.164_T to BTI_T and matches with ER_T.
GATESETUP
"Establishes the gate" for this call.
GATESETUPACK
SETUP
SETUPACK
GCSETUPACK
GATESETUP
"Establishes the gate" for this call.
GATESETUPACK
SETUPACK
Additional Capability Negotiation (if necessary)
Intermediate Routers
RESERVE
RESERVE
Reserves Backbone Channel
Reserves access channel
Reserves access channel
RESERVEACK
RESERVEACK
RING
RINGBACK
Rings phone
Starts ringback
CONNECT
Detects off-hook
COMMIT
COMMIT
Allocates access channel.
GATEOPEN
Allocates access channel.
Generates call record.
GATEOPEN
Generates call record.
COMMITACK
GATEOPENACK
GATEOPENACK
COMMITACK
Call In Progress

Figure 6

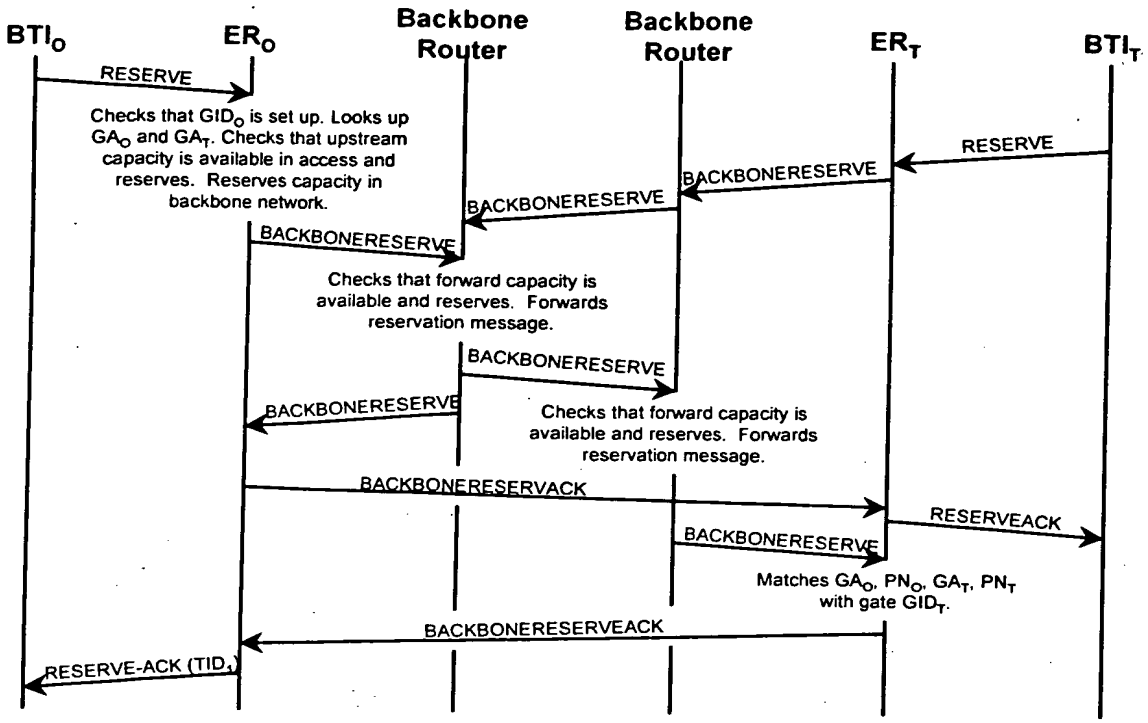
BTI$_O$
ER$_O$
Backbone Router
Backbone Router
ER$_T$
BTI$_T$
RESERVE
Checks that GID$_O$ is set up. Looks up GA$_O$ and GA$_T$. Checks that upstream capacity is available in access and reserves. Reserves capacity in backbone network.
RESERVE
BACKBONERESERVE
BACKBONERESERVE
BACKBONERESERVE
Checks that forward capacity is available and reserves. Forwards reservation message.
BACKBONERESERVE
BACKBONERESERVE
Checks that forward capacity is available and reserves. Forwards reservation message.
BACKBONERESERVACK
RESERVEACK
BACKBONERESERVE
Matches GA$_O$, PN$_O$, GA$_T$, PN$_T$ with gate GID$_T$.
BACKBONERESERVEACK
RESERVE-ACK (TID$_1$)

Figure 7

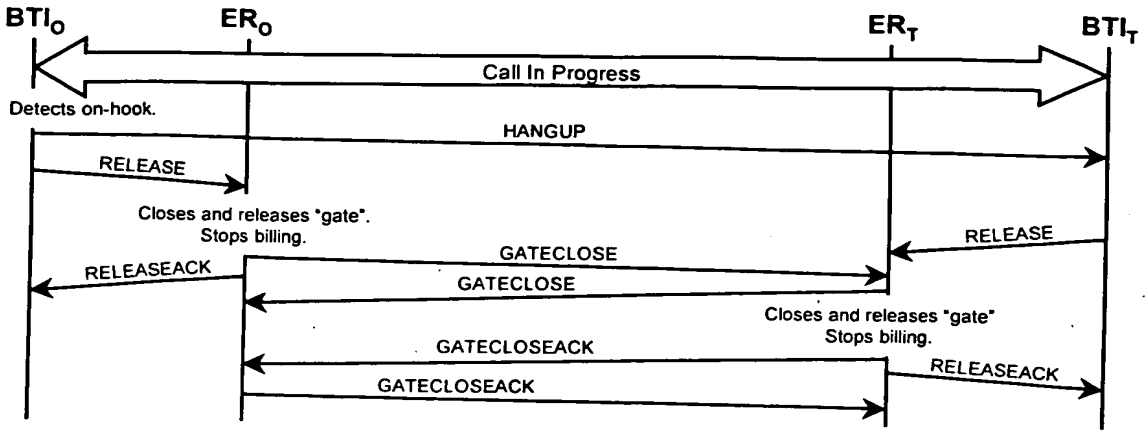
BTI$_O$
ER$_O$
ER$_T$
BTI$_T$
Call In Progress
Detects on-hook.
HANGUP
RELEASE
Closes and releases "gate".
Stops billing.
RELEASE
GATECLOSE
RELEASEACK
GATECLOSE
Closes and releases "gate"
Stops billing.
GATECLOSEACK
RELEASEACK
GATECLOSEACK

Figure 8

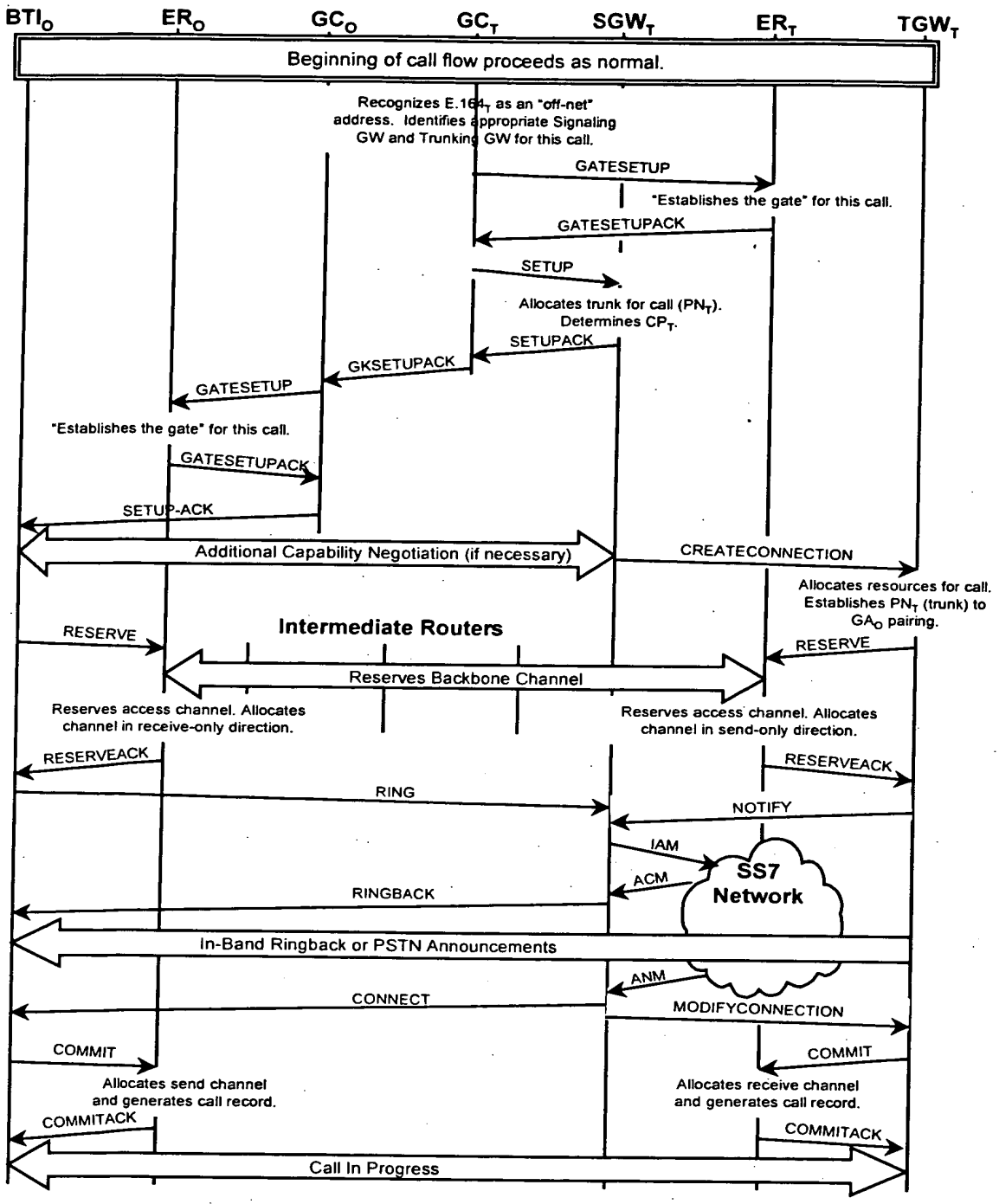
BTI$_O$
ER$_O$
GC$_O$
GC$_T$
SGW$_T$
ER$_T$
TGW$_T$
Beginning of call flow proceeds as normal.
Recognizes E.164$_T$ as an "off-net" address. Identifies appropriate Signaling GW and Trunking GW for this call.
GATESETUP
"Establishes the gate" for this call.
GATESETUPACK
SETUP
Allocates trunk for call (PN$_T$). Determines CP$_T$.
SETUPACK
GKSETUPACK
GATESETUP
"Establishes the gate" for this call.
GATESETUPACK
SETUP-ACK
Additional Capability Negotiation (if necessary)
CREATECONNECTION
Allocates resources for call. Establishes PN$_T$ (trunk) to GA$_O$ pairing.
RESERVE
Intermediate Routers
RESERVE
Reserves Backbone Channel
Reserves access channel. Allocates channel in receive-only direction.
Reserves access channel. Allocates channel in send-only direction.
RESERVEACK
RESERVEACK
RING
NOTIFY
IAM
ACM
SS7 Network
RINGBACK
In-Band Ringback or PSTN Announcements
ANM
CONNECT
MODIFYCONNECTION
COMMIT
COMMIT
Allocates send channel and generates call record.
Allocates receive channel and generates call record.
COMMITACK
COMMITACK
Call In Progress

Figure 9

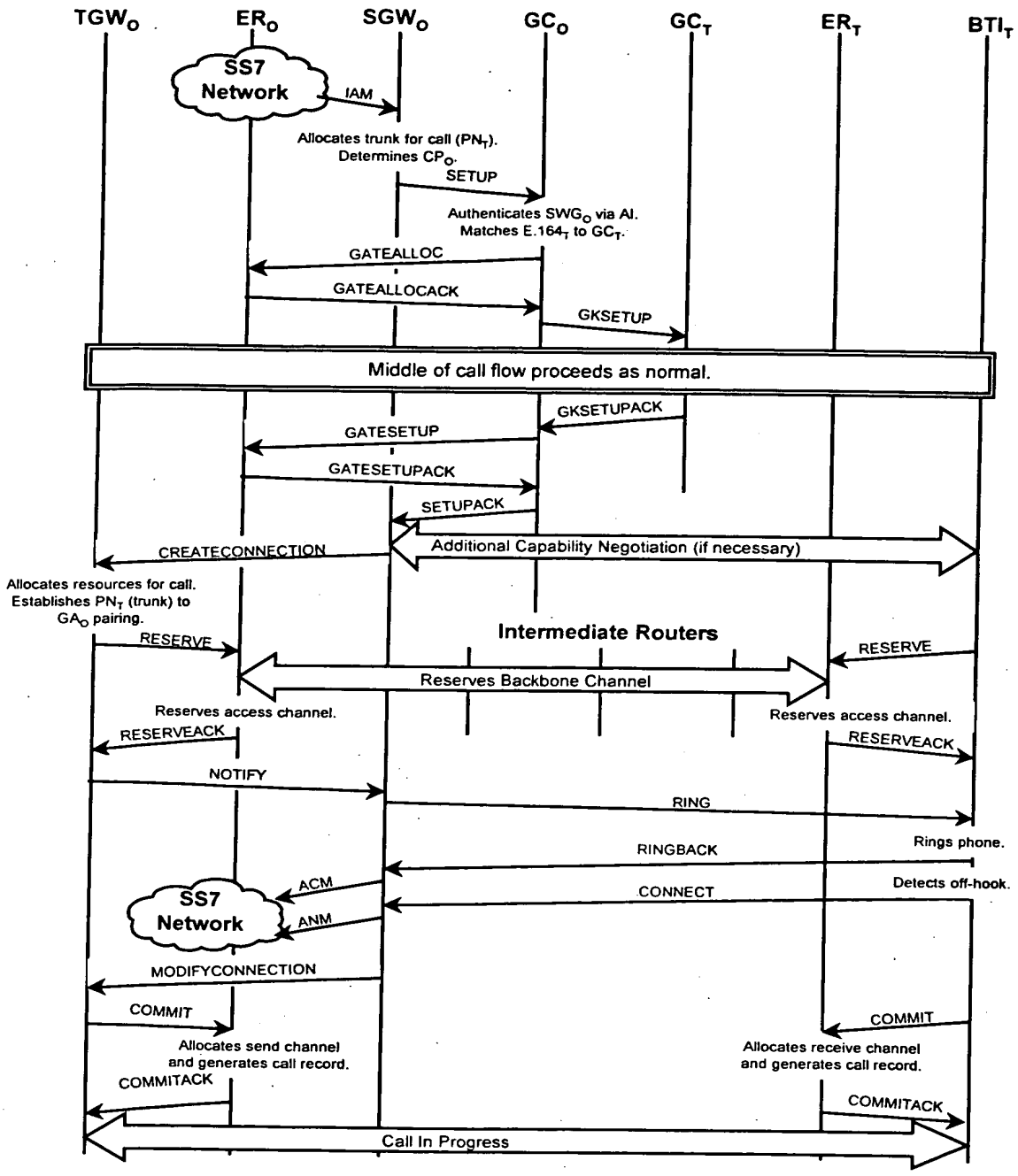
TGW$_O$
ER$_O$
SGW$_O$
GC$_O$
GC$_T$
ER$_T$
BTI$_T$
SS7 Network
IAM
Allocates trunk for call (PN$_T$).
Determines CP$_O$.
SETUP
Authenticates SWG$_O$ via AI.
Matches E.164$_T$ to GC$_T$.
GATEALLOC
GATEALLOCACK
GKSETUP
Middle of call flow proceeds as normal.
GKSETUPACK
GATESETUP
GATESETUPACK
SETUPACK
CREATECONNECTION
Additional Capability Negotiation (if necessary)
Allocates resources for call.
Establishes PN$_T$ (trunk) to GA$_O$ pairing.
RESERVE
Intermediate Routers
RESERVE
Reserves Backbone Channel
Reserves access channel.
Reserves access channel.
RESERVEACK
RESERVEACK
NOTIFY
RING
Rings phone.
RINGBACK
ACM
CONNECT
Detects off-hook.
SS7 Network
ANM
MODIFYCONNECTION
COMMIT
COMMIT
Allocates send channel and generates call record.
Allocates receive channel and generates call record.
COMMITACK
COMMITACK
Call In Progress

Figure 10

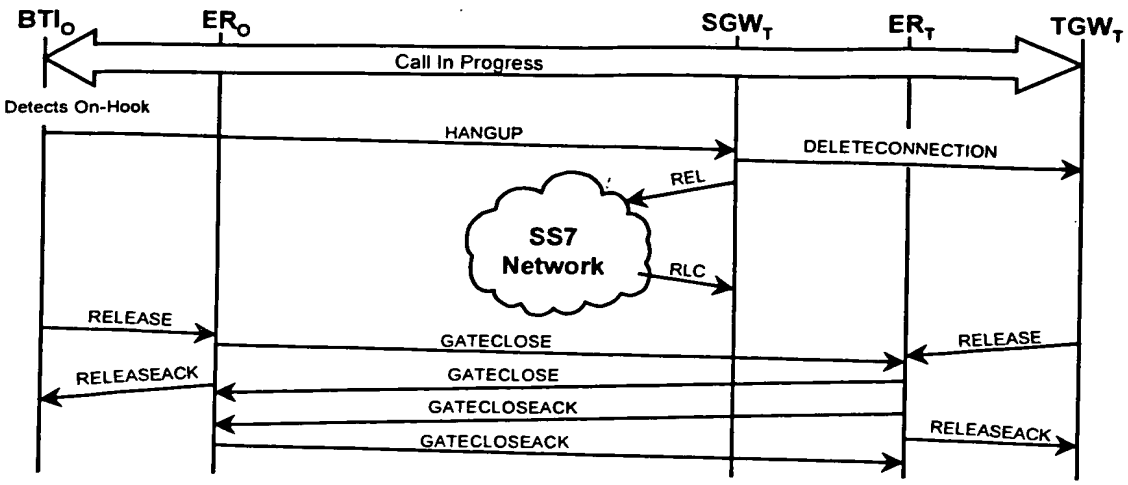

Figure 11

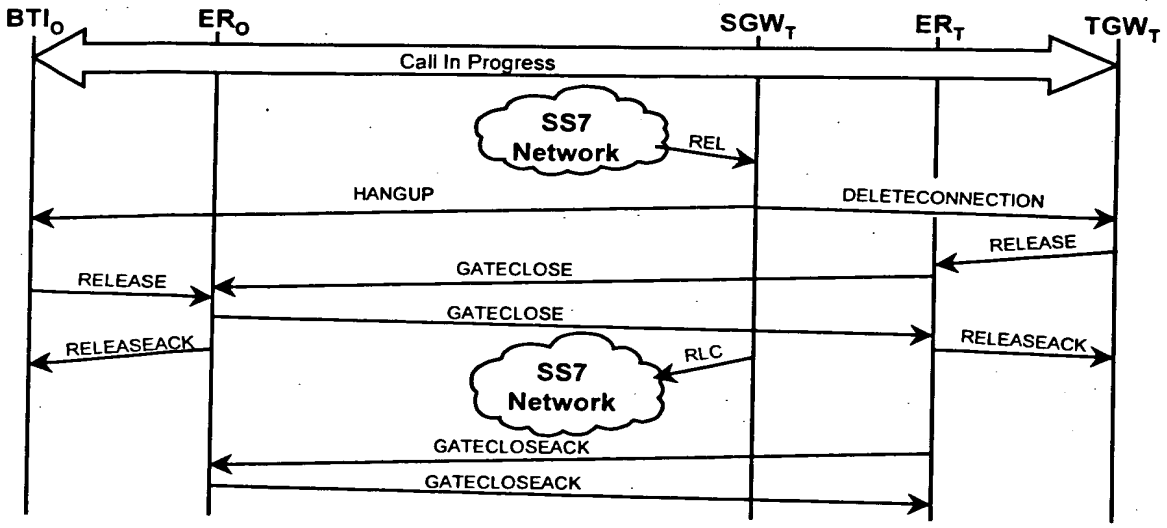

Figure 12

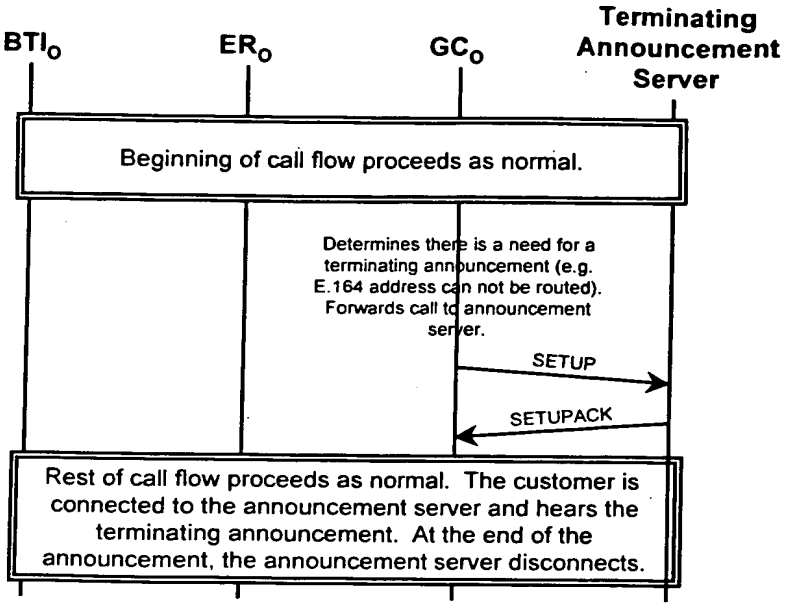

Figure 13

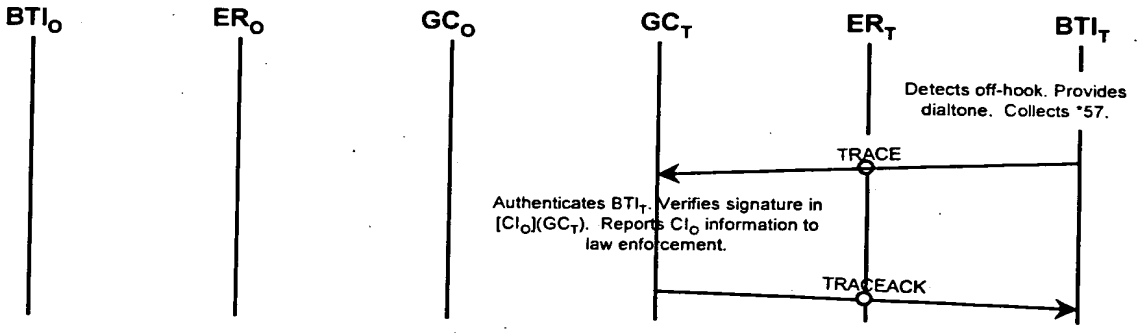

Figure 14

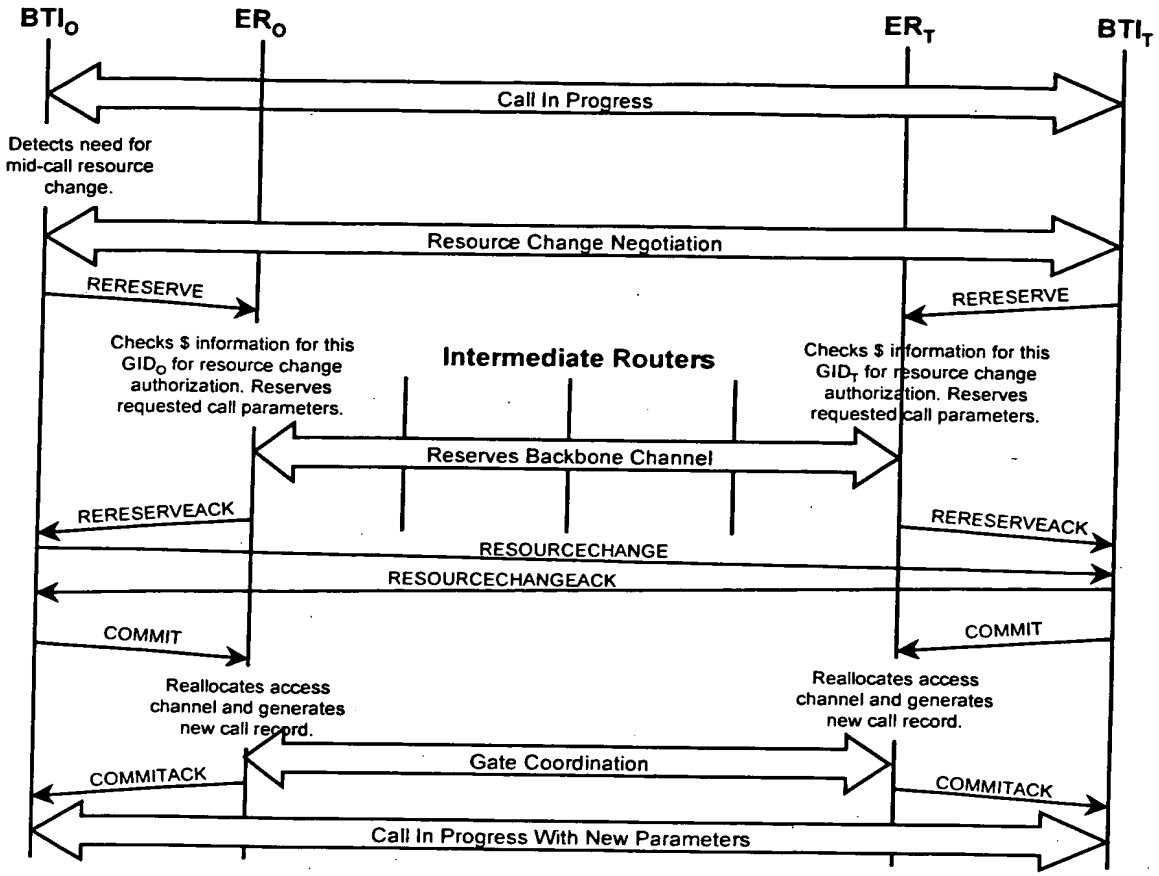

Figure 15

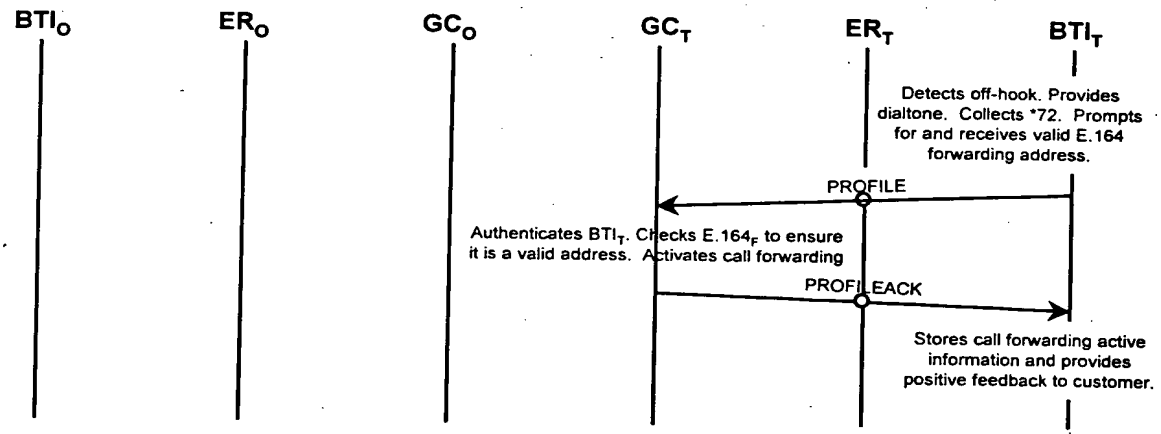

Figure 16

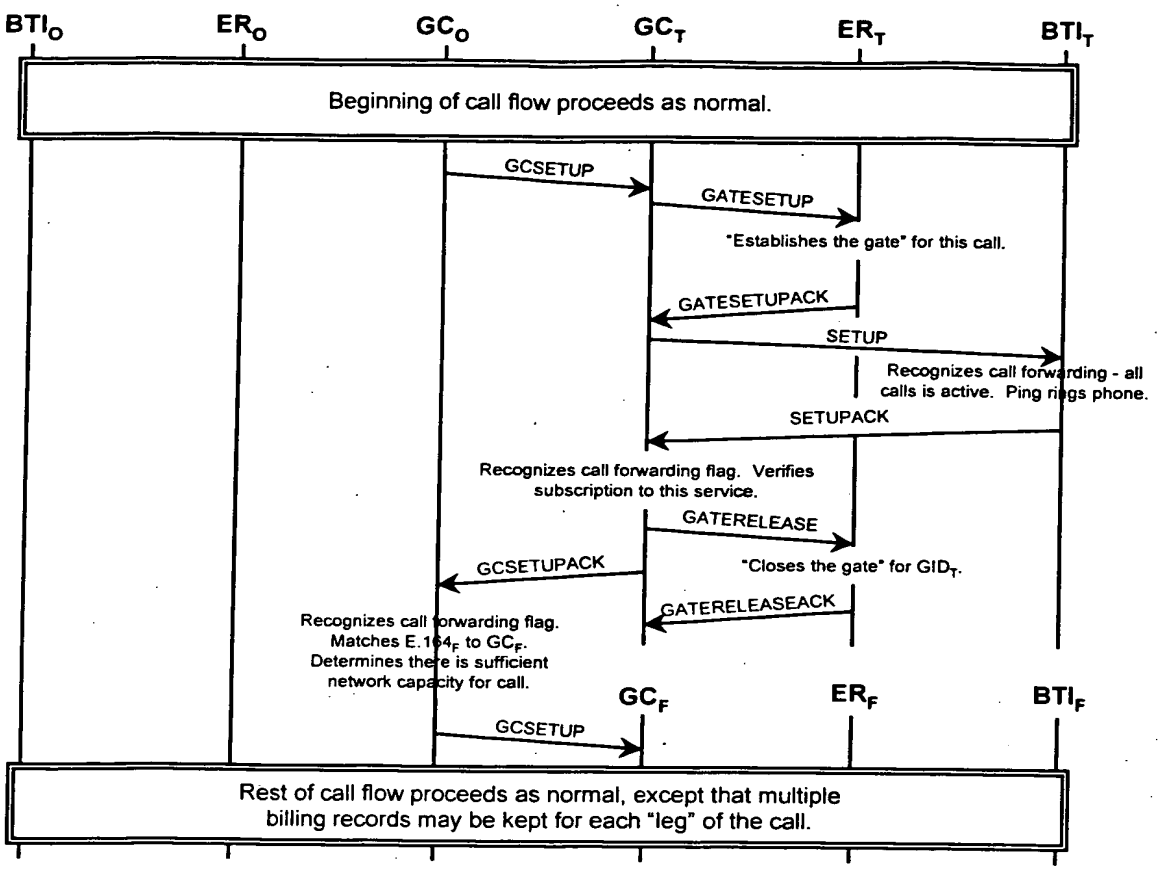
BTI_O
ER_O
GC_O
GC_T
ER_T
BTI_T
Beginning of call flow proceeds as normal.
GCSETUP
GATESETUP
"Establishes the gate" for this call.
GATESETUPACK
SETUP
Recognizes call forwarding - all calls is active. Ping rings phone.
SETUPACK
Recognizes call forwarding flag. Verifies subscription to this service.
GATERELEASE
GCSETUPACK
"Closes the gate" for GID_T.
GATERELEASEACK
Recognizes call forwarding flag. Matches E.164_F to GC_F. Determines there is sufficient network capacity for call.
GC_F
ER_F
BTI_F
GCSETUP
Rest of call flow proceeds as normal, except that multiple billing records may be kept for each "leg" of the call.

Figure 17

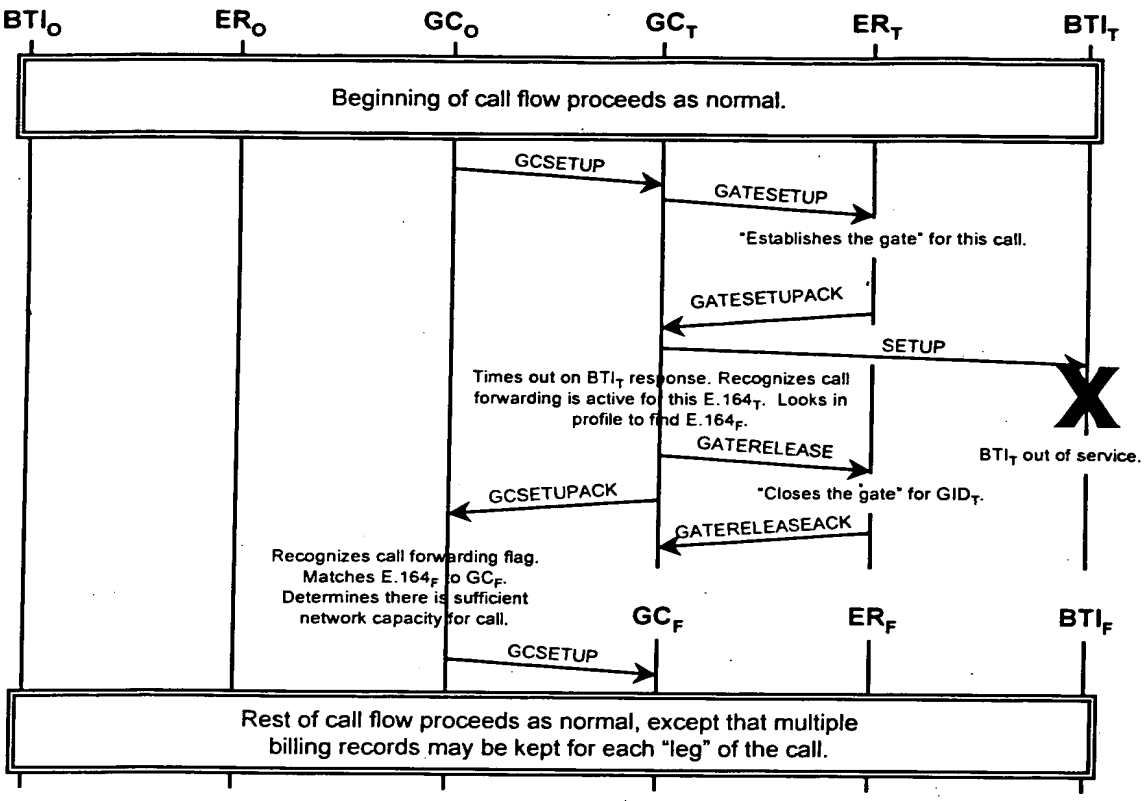
BTI_O
ER_O
GC_O
GC_T
ER_T
BTI_T
Beginning of call flow proceeds as normal.
GCSETUP
GATESETUP
"Establishes the gate" for this call.
GATESETUPACK
SETUP
Times out on BTI_T response. Recognizes call forwarding is active for this E.164_T. Looks in profile to find E.164_F.
GATERELEASE
BTI_T out of service.
GCSETUPACK
"Closes the gate" for GID_T.
GATERELEASEACK
Recognizes call forwarding flag. Matches E.164_F to GC_F. Determines there is sufficient network capacity for call.
GC_F
ER_F
BTI_F
GCSETUP
Rest of call flow proceeds as normal, except that multiple billing records may be kept for each "leg" of the call.

Figure 18

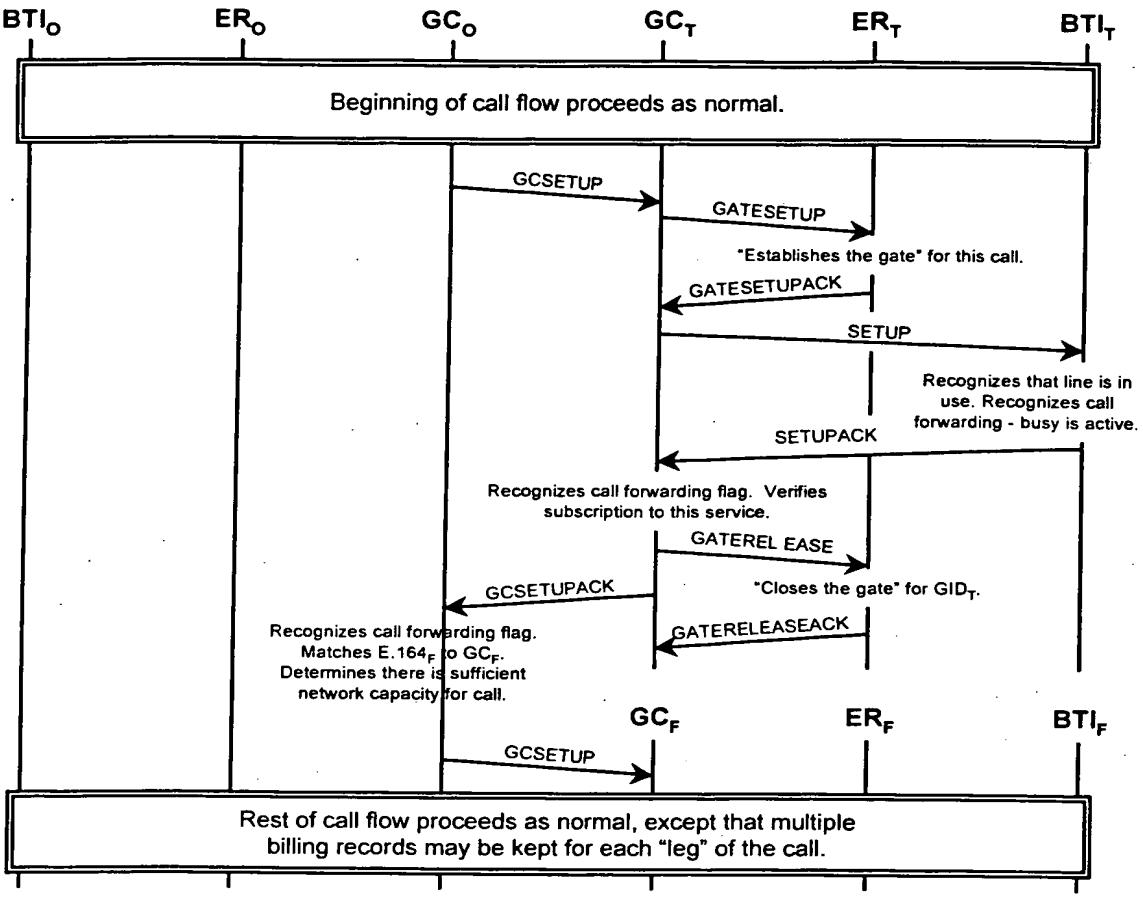
BTI$_O$
ER$_O$
GC$_O$
GC$_T$
ER$_T$
BTI$_T$
Beginning of call flow proceeds as normal.
GCSETUP
GATESETUP
"Establishes the gate" for this call.
GATESETUPACK
SETUP
Recognizes that line is in use. Recognizes call forwarding - busy is active.
SETUPACK
Recognizes call forwarding flag. Verifies subscription to this service.
GATEREL EASE
"Closes the gate" for GID$_T$.
GCSETUPACK
GATERELEASEACK
Recognizes call forwarding flag. Matches E.164$_F$ to GC$_F$. Determines there is sufficient network capacity for call.
GC$_F$
ER$_F$
BTI$_F$
GCSETUP
Rest of call flow proceeds as normal, except that multiple billing records may be kept for each "leg" of the call.

Figure 19

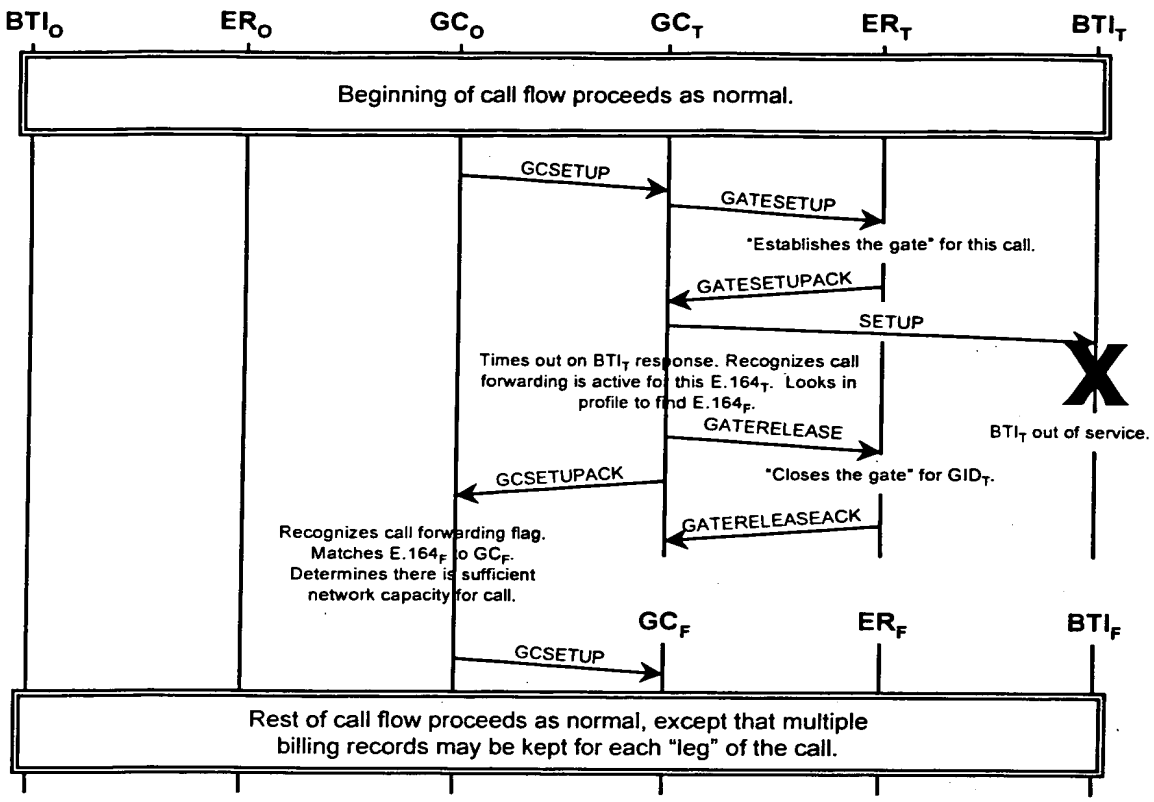
BTI_O
ER_O
GC_O
GC_T
ER_T
BTI_T
Beginning of call flow proceeds as normal.
GCSETUP
GATESETUP
"Establishes the gate" for this call.
GATESETUPACK
SETUP
Times out on BTI_T response. Recognizes call forwarding is active for this E.164_T. Looks in profile to find E.164_F.
GATERELEASE
BTI_T out of service.
"Closes the gate" for GID_T.
GCSETUPACK
GATERELEASEACK
Recognizes call forwarding flag. Matches E.164_F to GC_F. Determines there is sufficient network capacity for call.
GC_F
ER_F
BTI_F
GCSETUP
Rest of call flow proceeds as normal, except that multiple billing records may be kept for each "leg" of the call.

Figure 20

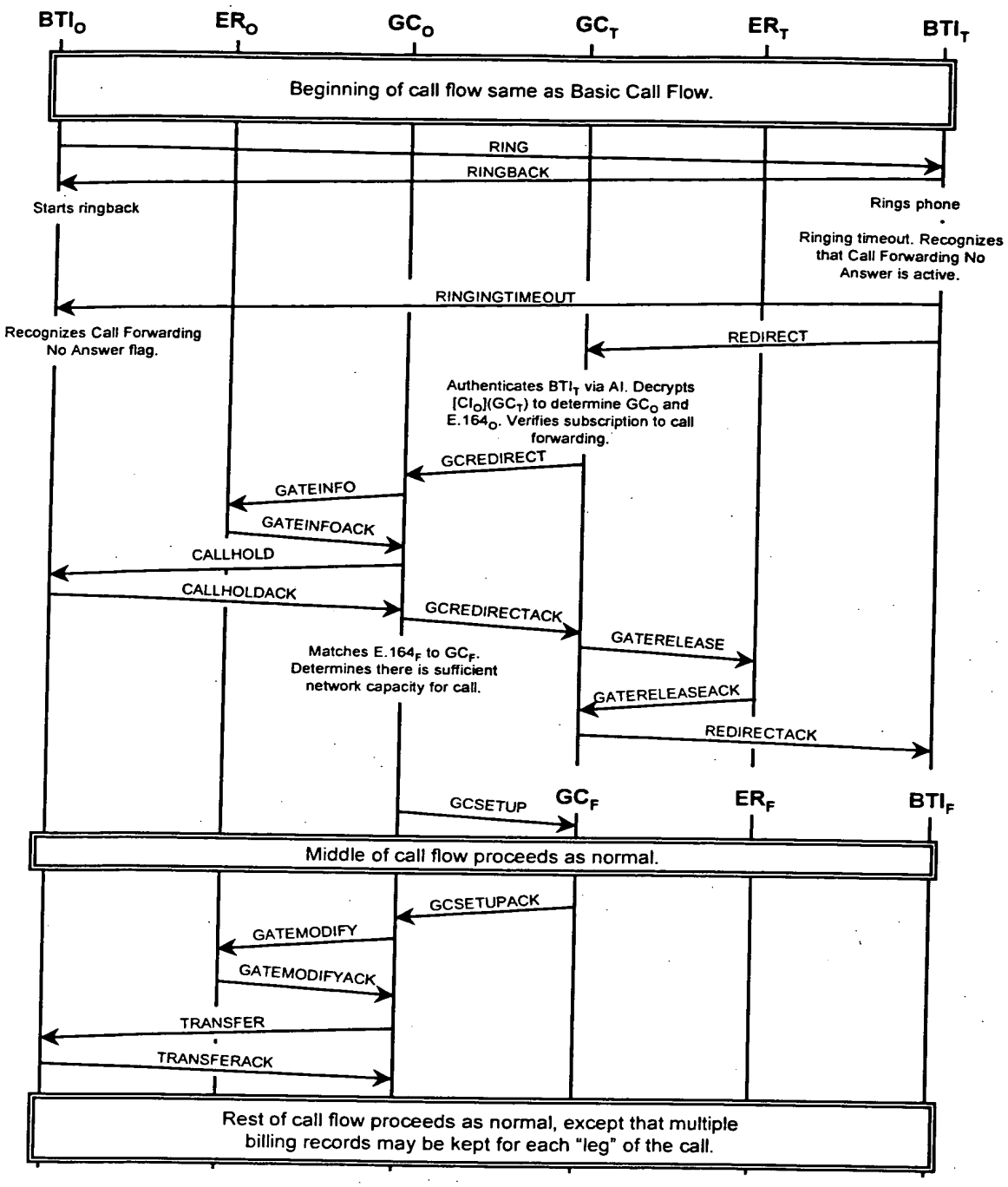
BTI_O
ER_O
GC_O
GC_T
ER_T
BTI_T
Beginning of call flow same as Basic Call Flow.
RING
RINGBACK
Starts ringback
Rings phone
Ringing timeout. Recognizes that Call Forwarding No Answer is active.
RINGINGTIMEOUT
Recognizes Call Forwarding No Answer flag.
REDIRECT
Authenticates BTI_T via AI. Decrypts [CI_O](GC_T) to determine GC_O and E.164_O. Verifies subscription to call forwarding.
GCREDIRECT
GATEINFO
GATEINFOACK
CALLHOLD
CALLHOLDACK
GCREDIRECTACK
GATERELEASE
Matches E.164_F to GC_F. Determines there is sufficient network capacity for call.
GATERELEASEACK
REDIRECTACK
GCSETUP
GC_F
ER_F
BTI_F
Middle of call flow proceeds as normal.
GCSETUPACK
GATEMODIFY
GATEMODIFYACK
TRANSFER
TRANSFERACK
Rest of call flow proceeds as normal, except that multiple billing records may be kept for each "leg" of the call.

Figure 21

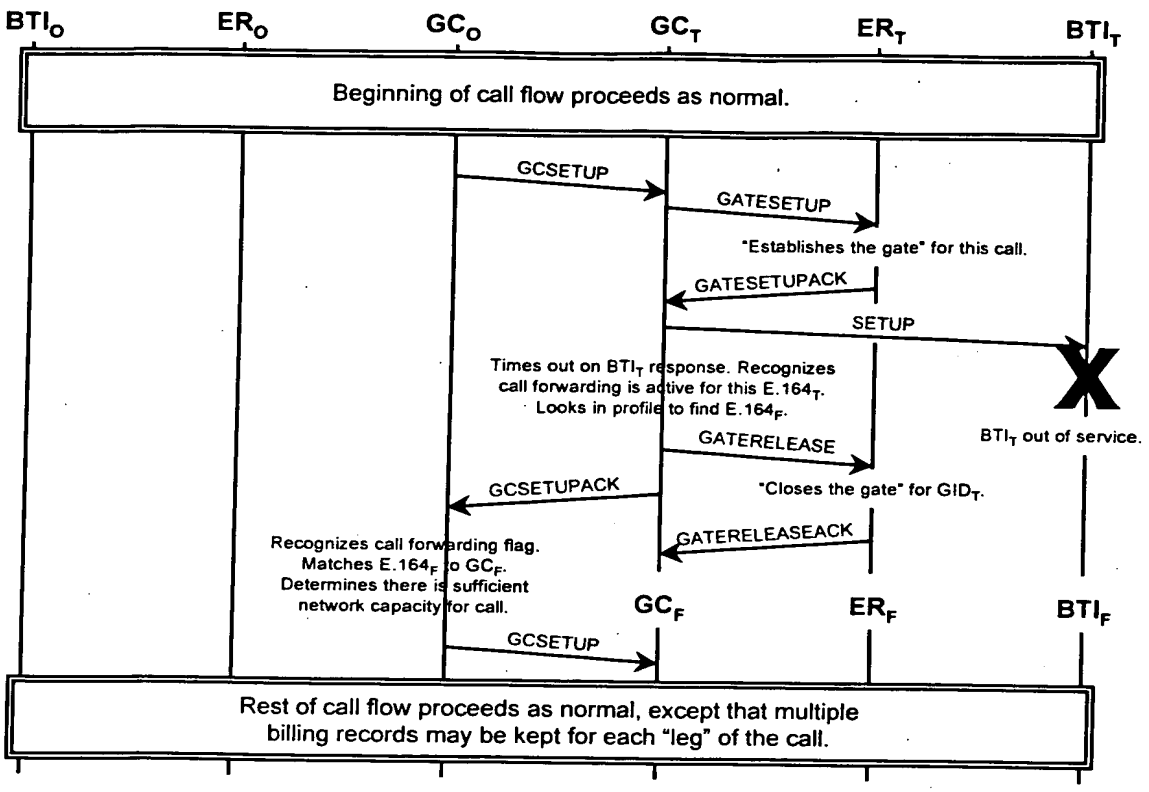
BTI_O
ER_O
GC_O
GC_T
ER_T
BTI_T
Beginning of call flow proceeds as normal.
GCSETUP
GATESETUP
"Establishes the gate" for this call.
GATESETUPACK
SETUP
Times out on BTI_T response. Recognizes call forwarding is active for this E.164_T. Looks in profile to find E.164_F.
BTI_T out of service.
GATERELEASE
GCSETUPACK
"Closes the gate" for GID_T.
GATERELEASEACK
Recognizes call forwarding flag. Matches E.164_F to GC_F. Determines there is sufficient network capacity for call.
GC_F
ER_F
BTI_F
GCSETUP
Rest of call flow proceeds as normal, except that multiple billing records may be kept for each "leg" of the call.

Figure 22

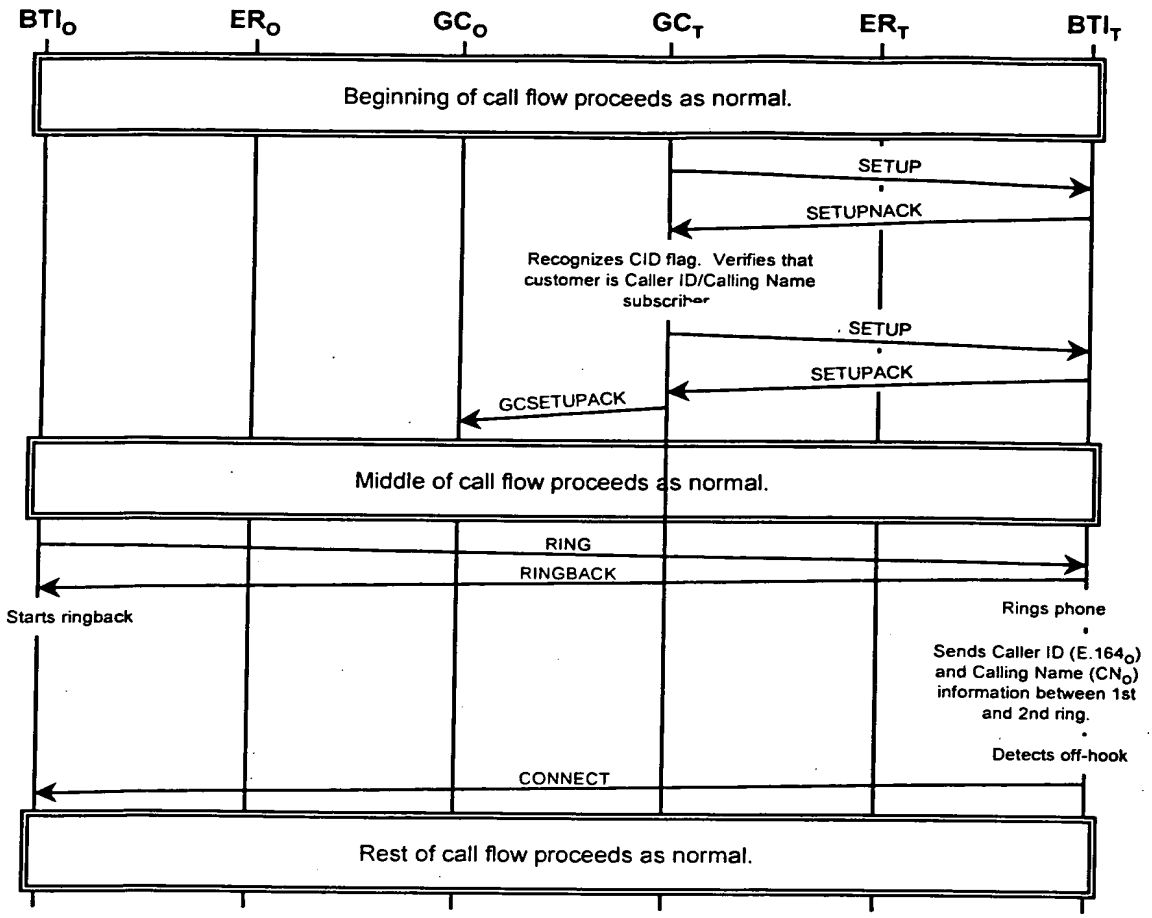
BTI$_O$
ER$_O$
GC$_O$
GC$_T$
ER$_T$
BTI$_T$
Beginning of call flow proceeds as normal.
SETUP
SETUPNACK
Recognizes CID flag. Verifies that customer is Caller ID/Calling Name subscriber
SETUP
SETUPACK
GCSETUPACK
Middle of call flow proceeds as normal.
RING
RINGBACK
Starts ringback
Rings phone
Sends Caller ID (E.164$_O$) and Calling Name (CN$_O$) information between 1st and 2nd ring.
Detects off-hook
CONNECT
Rest of call flow proceeds as normal.

Figure 23

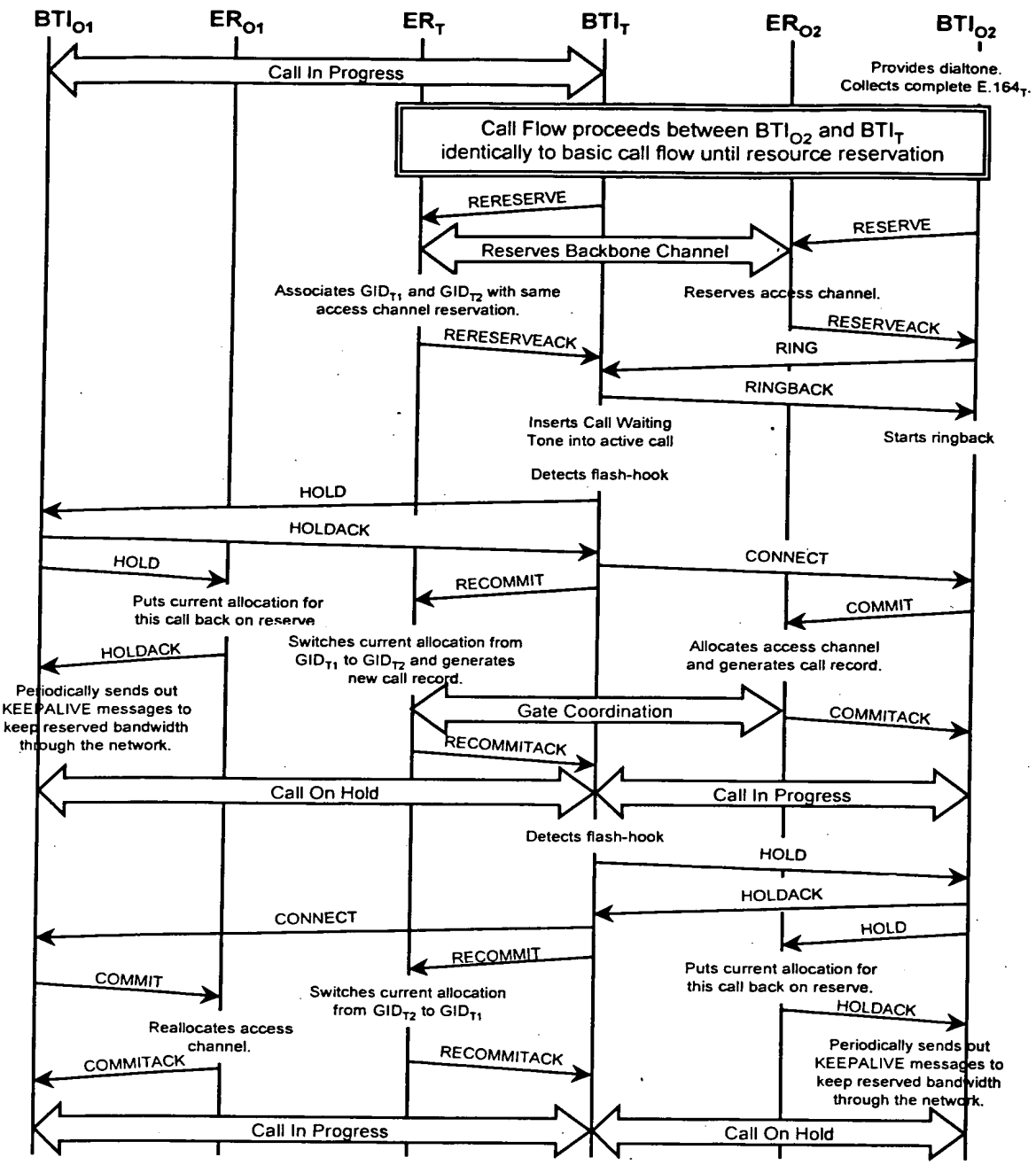
BTI$_{O1}$
ER$_{O1}$
ER$_{T}$
BTI$_{T}$
ER$_{O2}$
BTI$_{O2}$
Call In Progress
Provides dialtone.
Collects complete E.164$_{T}$.
Call Flow proceeds between BTI$_{O2}$ and BTI$_{T}$ identically to basic call flow until resource reservation
RERESERVE
RESERVE
Reserves Backbone Channel
Associates GID$_{T1}$ and GID$_{T2}$ with same access channel reservation.
Reserves access channel.
RESERVEACK
RERESERVEACK
RING
RINGBACK
Inserts Call Waiting Tone into active call
Starts ringback
Detects flash-hook
HOLD
HOLDACK
HOLD
CONNECT
RECOMMIT
COMMIT
Puts current allocation for this call back on reserve
HOLDACK
Switches current allocation from GID$_{T1}$ to GID$_{T2}$ and generates new call record.
Allocates access channel and generates call record.
Periodically sends out KEEPALIVE messages to keep reserved bandwidth through the network.
Gate Coordination
COMMITACK
RECOMMITACK
Call On Hold
Call In Progress
Detects flash-hook
HOLD
HOLDACK
CONNECT
HOLD
RECOMMIT
COMMIT
Puts current allocation for this call back on reserve.
Switches current allocation from GID$_{T2}$ to GID$_{T1}$
HOLDACK
Reallocates access channel.
RECOMMITACK
COMMITACK
Periodically sends out KEEPALIVE messages to keep reserved bandwidth through the network.
Call In Progress
Call On Hold

Figure 24

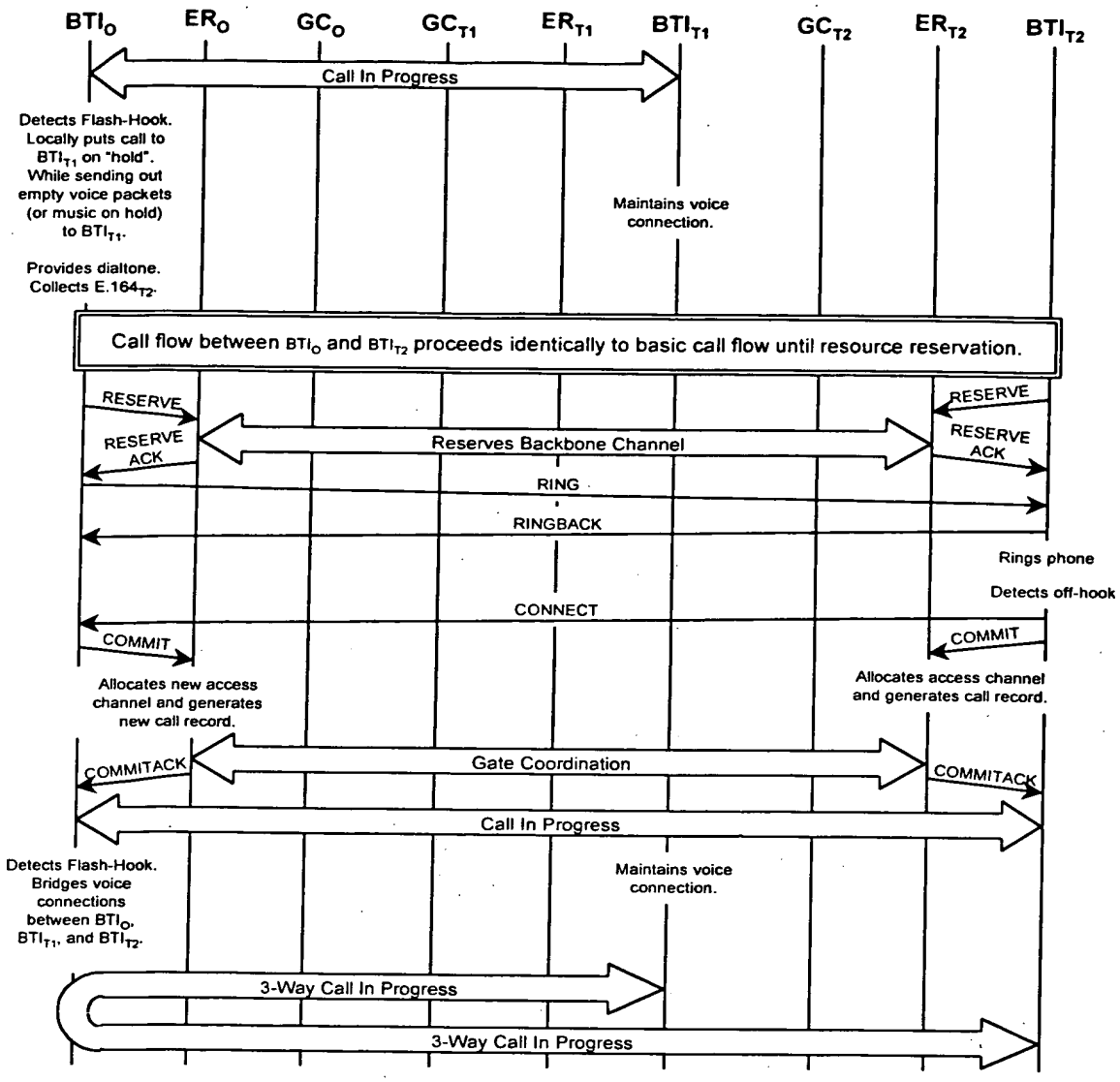
BTI_O
ER_O
GC_O
GC_T1
ER_T1
BTI_T1
GC_T2
ER_T2
BTI_T2
Call In Progress
Detects Flash-Hook. Locally puts call to BTI_T1 on "hold". While sending out empty voice packets (or music on hold) to BTI_T1.
Provides dialtone. Collects E.164_T2.
Maintains voice connection.
Call flow between BTI_O and BTI_T2 proceeds identically to basic call flow until resource reservation.
RESERVE
RESERVE ACK
Reserves Backbone Channel
RESERVE
RESERVE ACK
RING
RINGBACK
Rings phone
Detects off-hook
CONNECT
COMMIT
COMMIT
Allocates new access channel and generates new call record.
Allocates access channel and generates call record.
COMMITACK
Gate Coordination
COMMITACK
Call In Progress
Detects Flash-Hook. Bridges voice connections between BTI_O, BTI_T1, and BTI_T2.
Maintains voice connection.
3-Way Call In Progress
3-Way Call In Progress

Figure 25

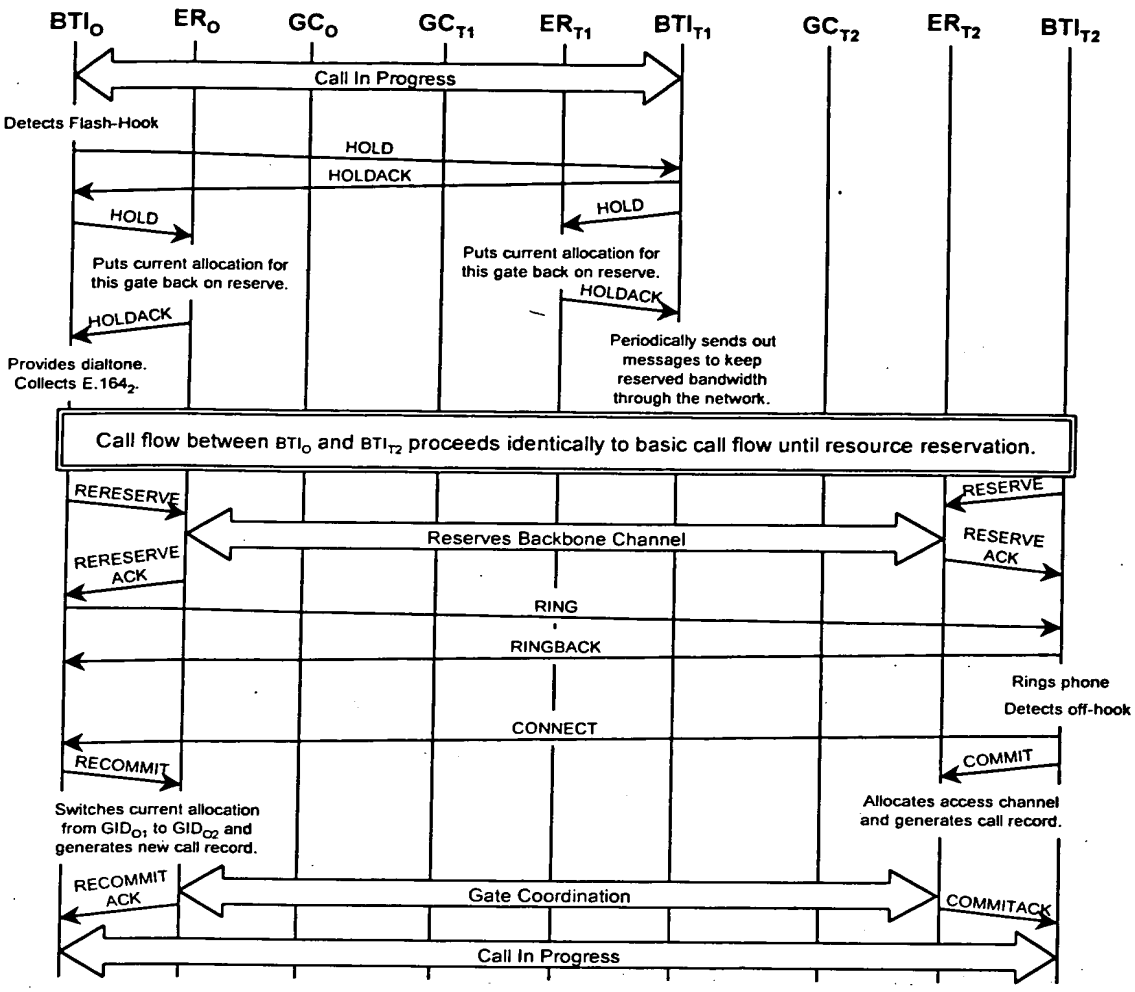
BTI$_O$
ER$_O$
GC$_O$
GC$_{T1}$
ER$_{T1}$
BTI$_{T1}$
GC$_{T2}$
ER$_{T2}$
BTI$_{T2}$
Call In Progress
Detects Flash-Hook
HOLD
HOLDACK
HOLD
HOLD
Puts current allocation for this gate back on reserve.
Puts current allocation for this gate back on reserve.
HOLDACK
HOLDACK
Provides dialtone. Collects E.164$_2$.
Periodically sends out messages to keep reserved bandwidth through the network.
Call flow between BTI$_O$ and BTI$_{T2}$ proceeds identically to basic call flow until resource reservation.
RERESERVE
RESERVE
Reserves Backbone Channel
RERESERVE ACK
RESERVE ACK
RING
RINGBACK
Rings phone
Detects off-hook
CONNECT
RECOMMIT
COMMIT
Switches current allocation from GID$_{O1}$ to GID$_{O2}$ and generates new call record.
Allocates access channel and generates call record.
RECOMMIT ACK
Gate Coordination
COMMITACK
Call In Progress

Figure 26

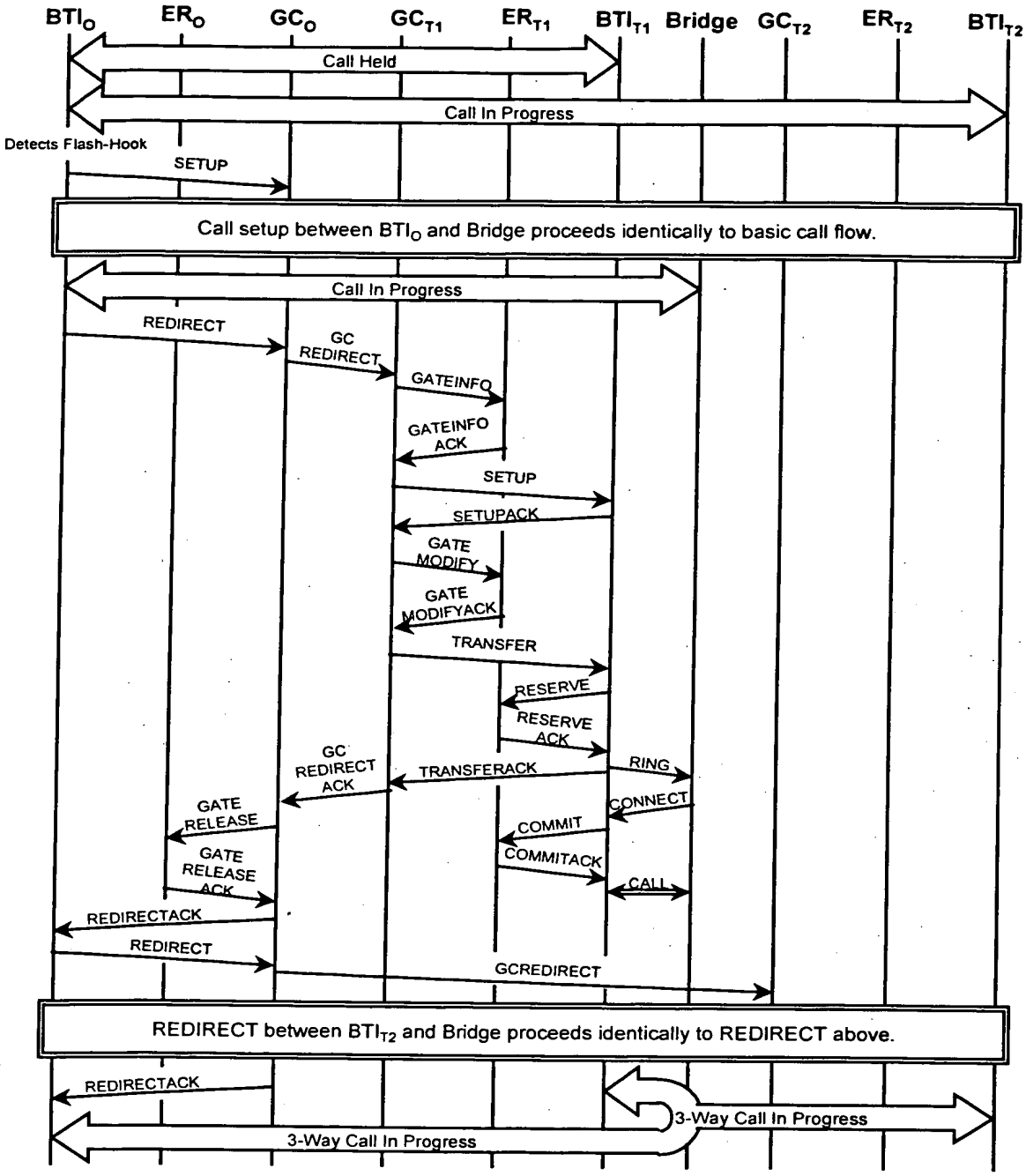
BTI$_O$
ER$_O$
GC$_O$
GC$_{T1}$
ER$_{T1}$
BTI$_{T1}$
Bridge
GC$_{T2}$
ER$_{T2}$
BTI$_{T2}$
Call Held
Call In Progress
Detects Flash-Hook
SETUP
Call setup between BTI$_O$ and Bridge proceeds identically to basic call flow.
Call In Progress
REDIRECT
GC REDIRECT
GATEINFO
GATEINFO ACK
SETUP
SETUPACK
GATE MODIFY
GATE MODIFYACK
TRANSFER
RESERVE
RESERVE ACK
GC REDIRECT ACK
TRANSFERACK
RING
CONNECT
GATE RELEASE
COMMIT
COMMITACK
GATE RELEASE ACK
CALL
REDIRECTACK
REDIRECT
GCREDIRECT
REDIRECT between BTI$_{T2}$ and Bridge proceeds identically to REDIRECT above.
REDIRECTACK
3-Way Call In Progress
3-Way Call In Progress

Figure 27

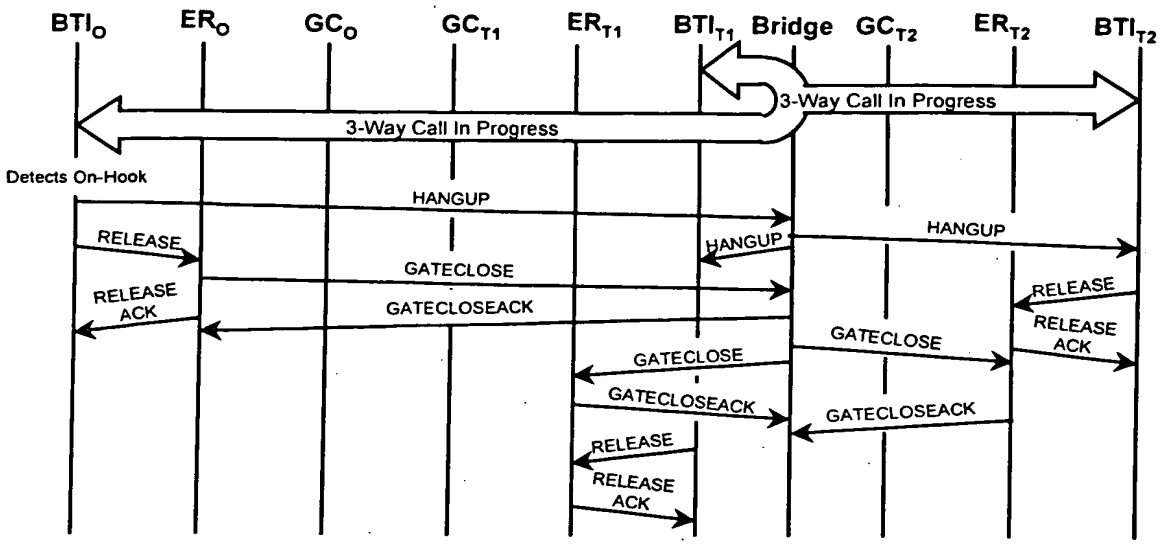
BTI$_O$
ER$_O$
GC$_O$
GC$_{T1}$
ER$_{T1}$
BTI$_{T1}$
Bridge
GC$_{T2}$
ER$_{T2}$
BTI$_{T2}$
3-Way Call In Progress
3-Way Call In Progress
Detects On-Hook
HANGUP
RELEASE
HANGUP
HANGUP
GATECLOSE
RELEASE
RELEASE ACK
GATECLOSEACK
RELEASE
ACK
GATECLOSE
GATECLOSE
GATECLOSEACK
GATECLOSEACK
RELEASE
RELEASE ACK

Figure 28

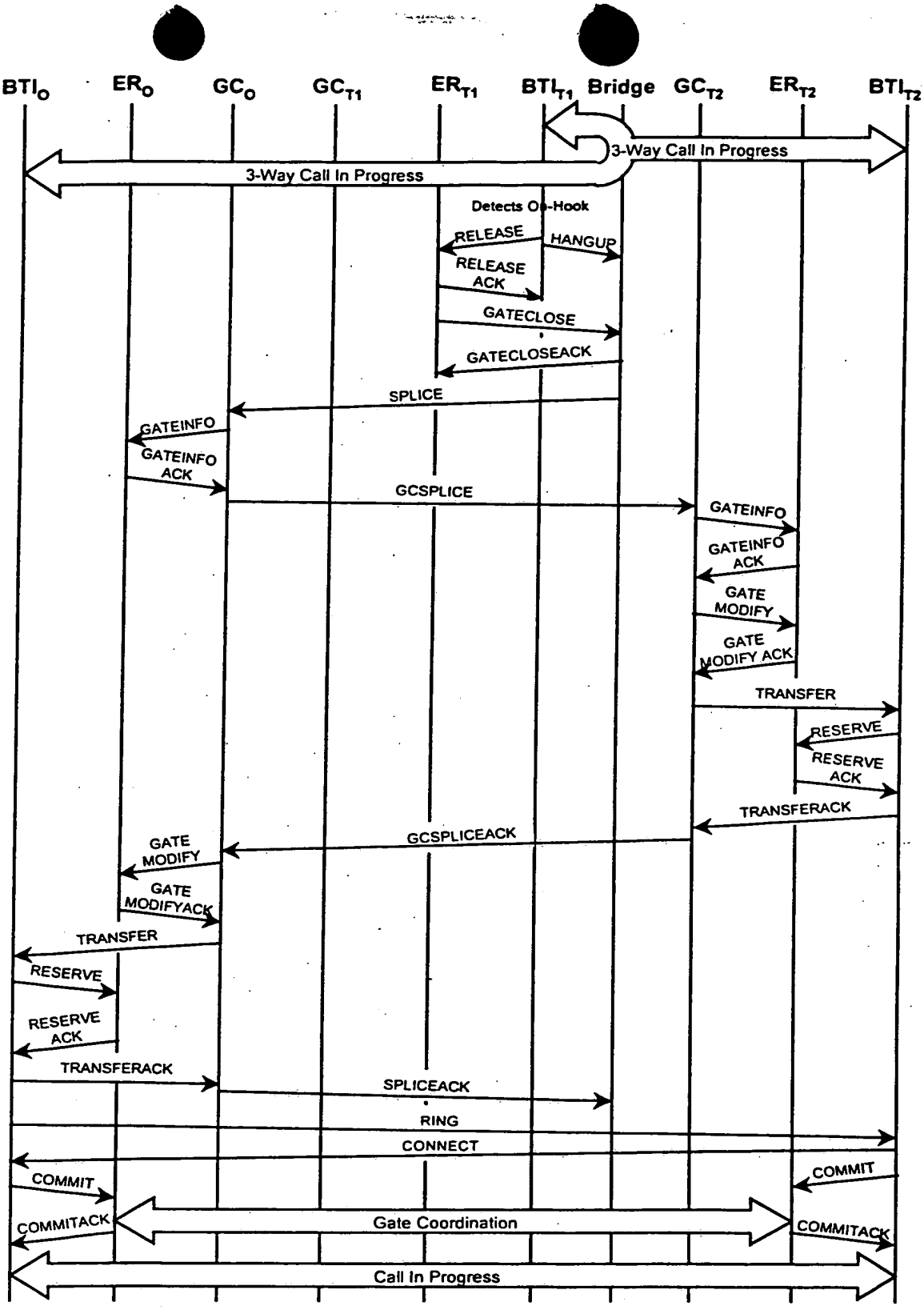
BTI_O
ER_O
GC_O
GC_T1
ER_T1
BTI_T1
Bridge
GC_T2
ER_T2
BTI_T2
3-Way Call In Progress
3-Way Call In Progress
Detects On-Hook
RELEASE
HANGUP
RELEASE ACK
GATECLOSE
GATECLOSEACK
SPLICE
GATEINFO
GATEINFO ACK
GCSPLICE
GATEINFO
GATEINFO ACK
GATE MODIFY
GATE MODIFY ACK
TRANSFER
RESERVE
RESERVE ACK
TRANSFERACK
GCSPLICEACK
GATE MODIFY
GATE MODIFYACK
TRANSFER
RESERVE
RESERVE ACK
TRANSFERACK
SPLICEACK
RING
CONNECT
COMMIT
COMMIT
COMMITACK
Gate Coordination
COMMITACK
Call In Progress

Figure 29

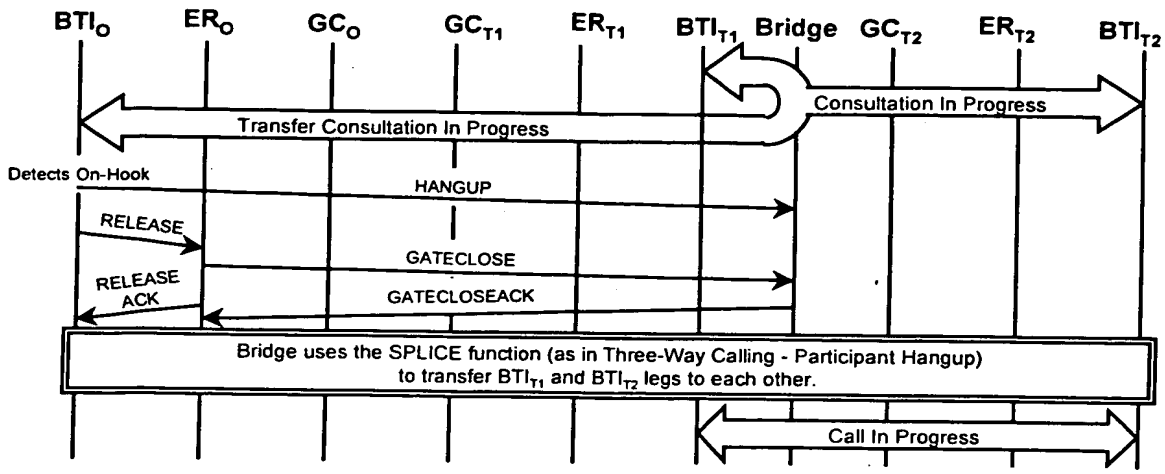
BTI$_O$
ER$_O$
GC$_O$
GC$_{T1}$
ER$_{T1}$
BTI$_{T1}$
Bridge
GC$_{T2}$
ER$_{T2}$
BTI$_{T2}$
Consultation In Progress
Transfer Consultation In Progress
Detects On-Hook
HANGUP
RELEASE
GATECLOSE
RELEASE ACK
GATECLOSEACK
Bridge uses the SPLICE function (as in Three-Way Calling - Participant Hangup)
to transfer BTI$_{T1}$ and BTI$_{T2}$ legs to each other.
Call In Progress

Figure 30

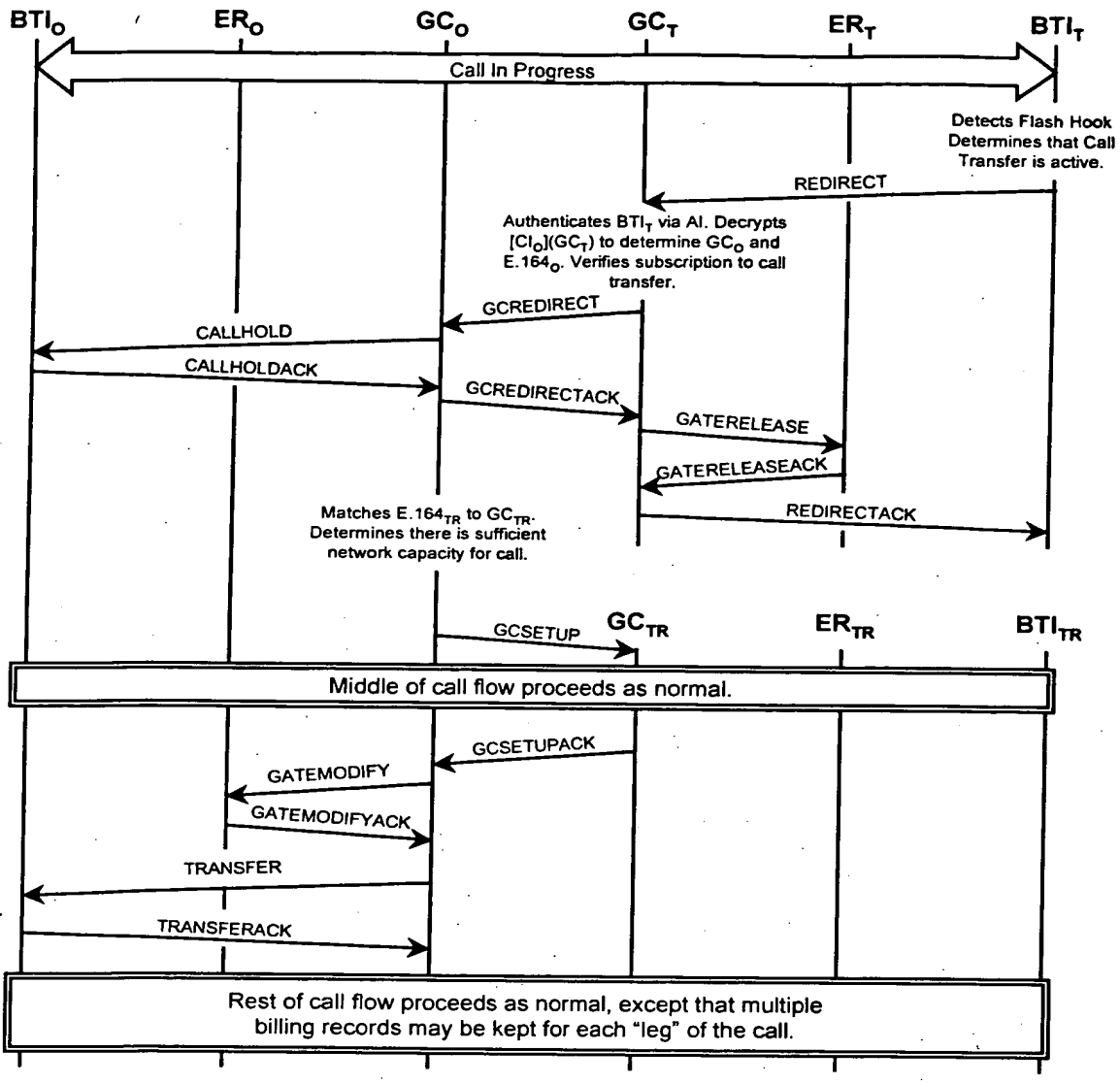

Figure 31

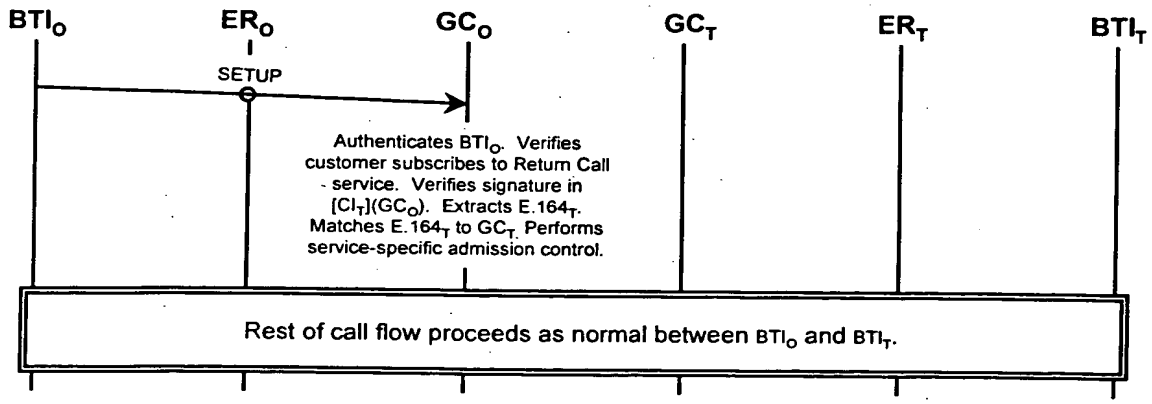

Figure 32

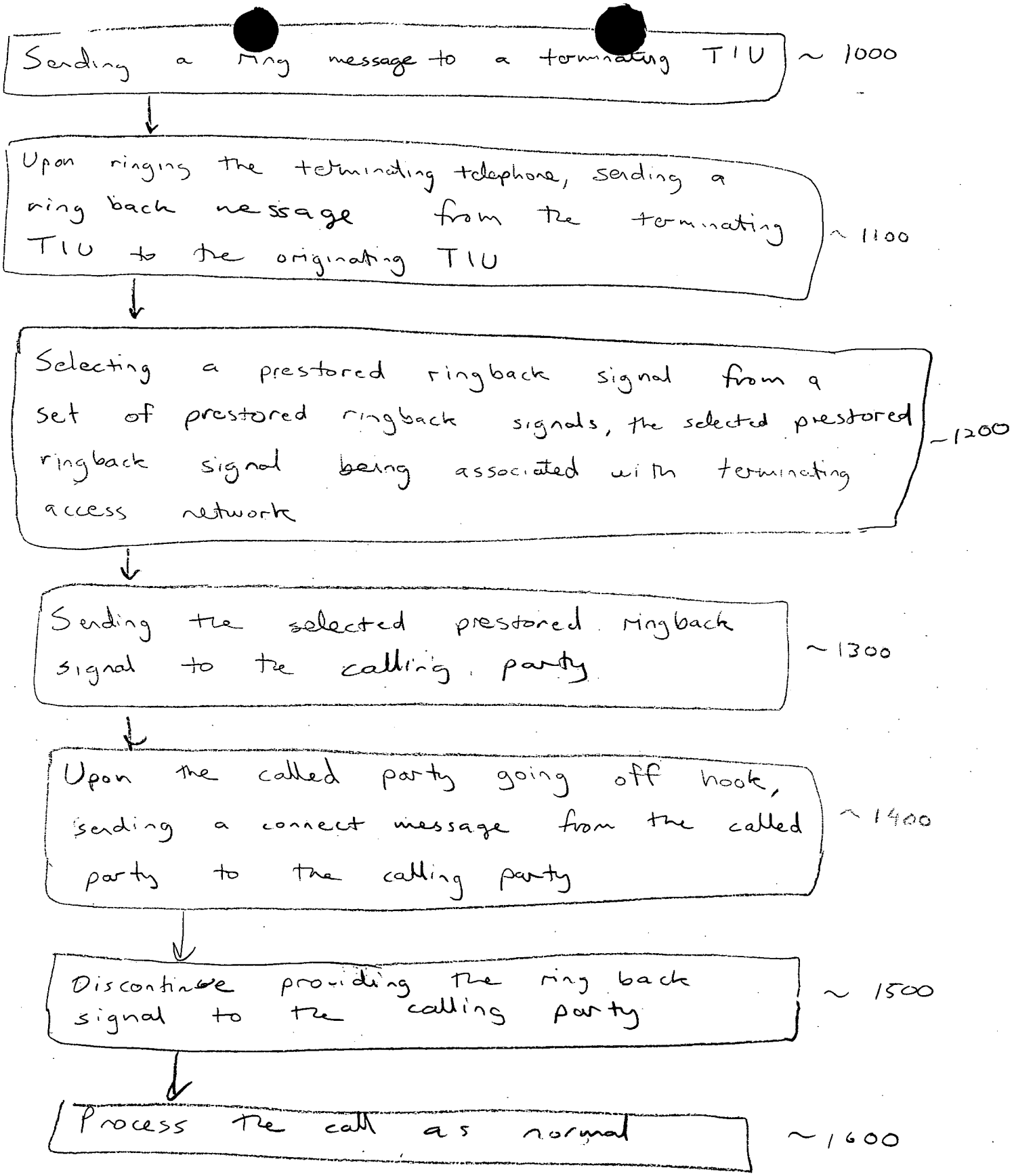
Sending a ring message to a terminating TIU
~ 1000
Upon ringing the terminating telephone, sending a ring back message from the terminating TIU to the originating TIU
~ 1100
Selecting a prestored ringback signal from a set of prestored ringback signals, the selected prestored ringback signal being associated with terminating access network
~ 1200
Sending the selected prestored ringback signal to the calling party
~ 1300
Upon the called party going off hook, sending a connect message from the called party to the calling party
~ 1400
Discontinue providing the ring back signal to the calling party
~ 1500
Process the call as normal
~ 1600

FIG. 33

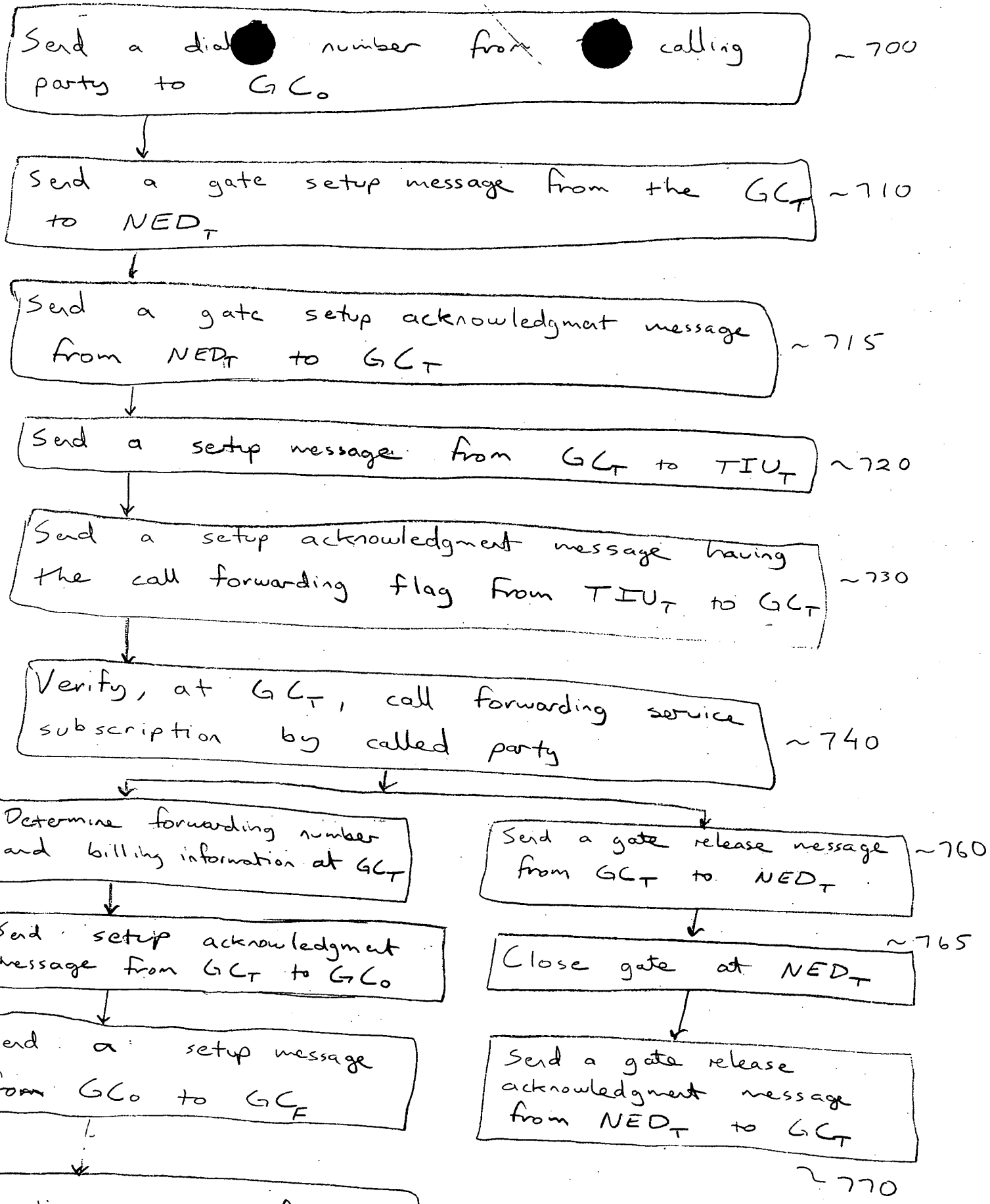
Send a dial number from calling party to GC_O
~700
Send a gate setup message from the GC_T to NED_T
~710
Send a gate setup acknowledgment message from NED_T to GC_T
~715
Send a setup message from GC_T to TIU_T
~720
Send a setup acknowledgment message having the call forwarding flag from TIU_T to GC_T
~730
Verify, at GC_T, call forwarding service subscription by called party
~740
Determine forwarding number and billing information at GC_T
Send setup acknowledgment message from GC_T to GC_O
Send a setup message from GC_O to GC_F
Continue processing of call
780
Send a gate release message from GC_T to NED_T
~760
Close gate at NED_T
~765
Send a gate release acknowledgment message from NED_T to GC_T
770

FIG. 34

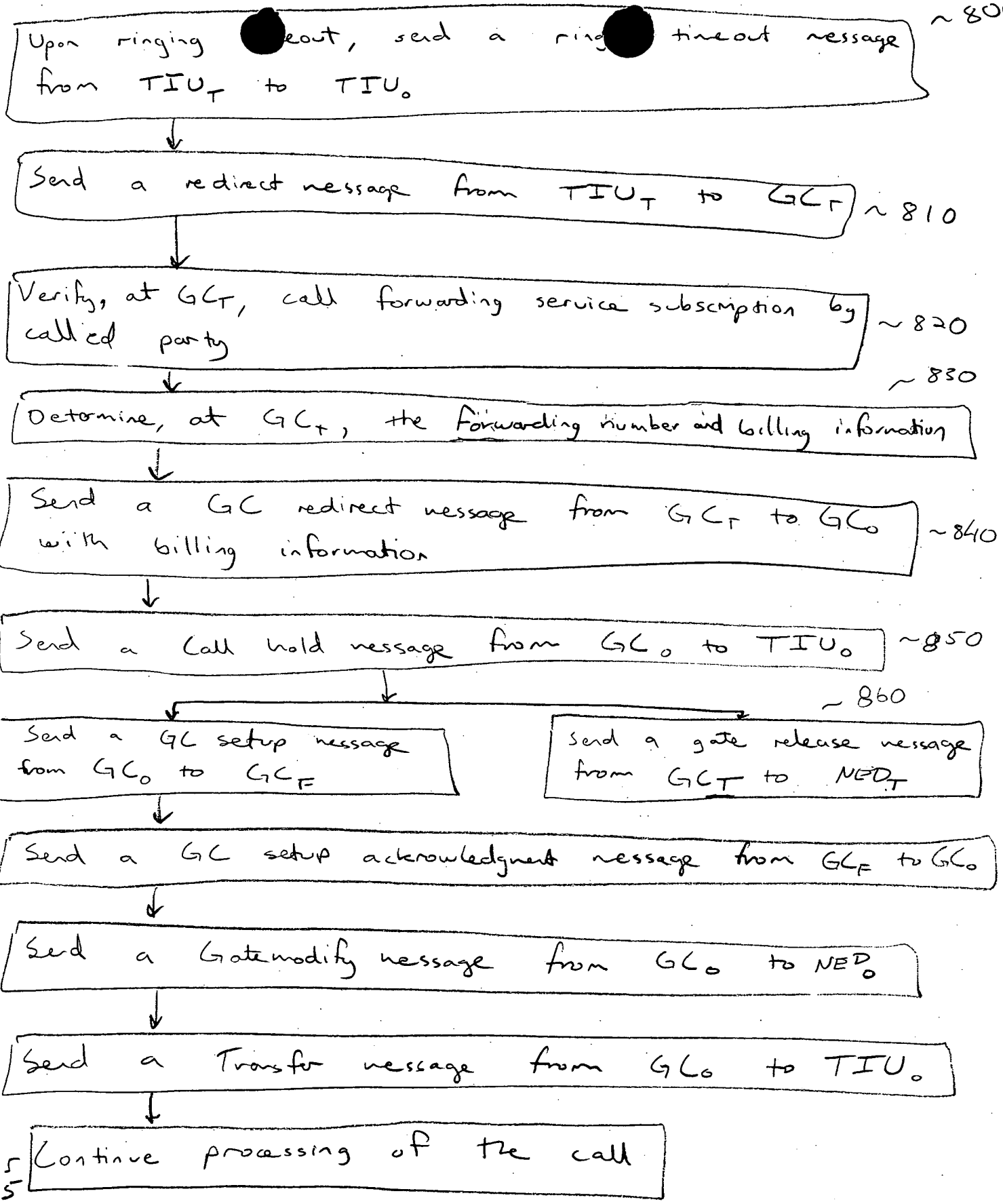
Upon ringing [redacted]eout, send a ring[redacted] timeout message from TIU_T to TIU_O
~800
Send a redirect message from TIU_T to GC_T
~810
Verify, at GC_T, call forwarding service subscription by called party
~820
Determine, at GC_T, the forwarding number and billing information
~830
Send a GC redirect message from GC_T to GC_O with billing information
~840
Send a Call hold message from GC_O to TIU_O
~850
Send a GC setup message from GC_O to GC_F
870
Send a gate release message from GC_T to NED_T
~860
Send a GC setup acknowledgment message from GC_F to GC_O
875
Send a Gatemodify message from GC_O to NED_O
880
Send a Transfer message from GC_O to TIU_O
890
Continue processing of the call
895

FIG. 35

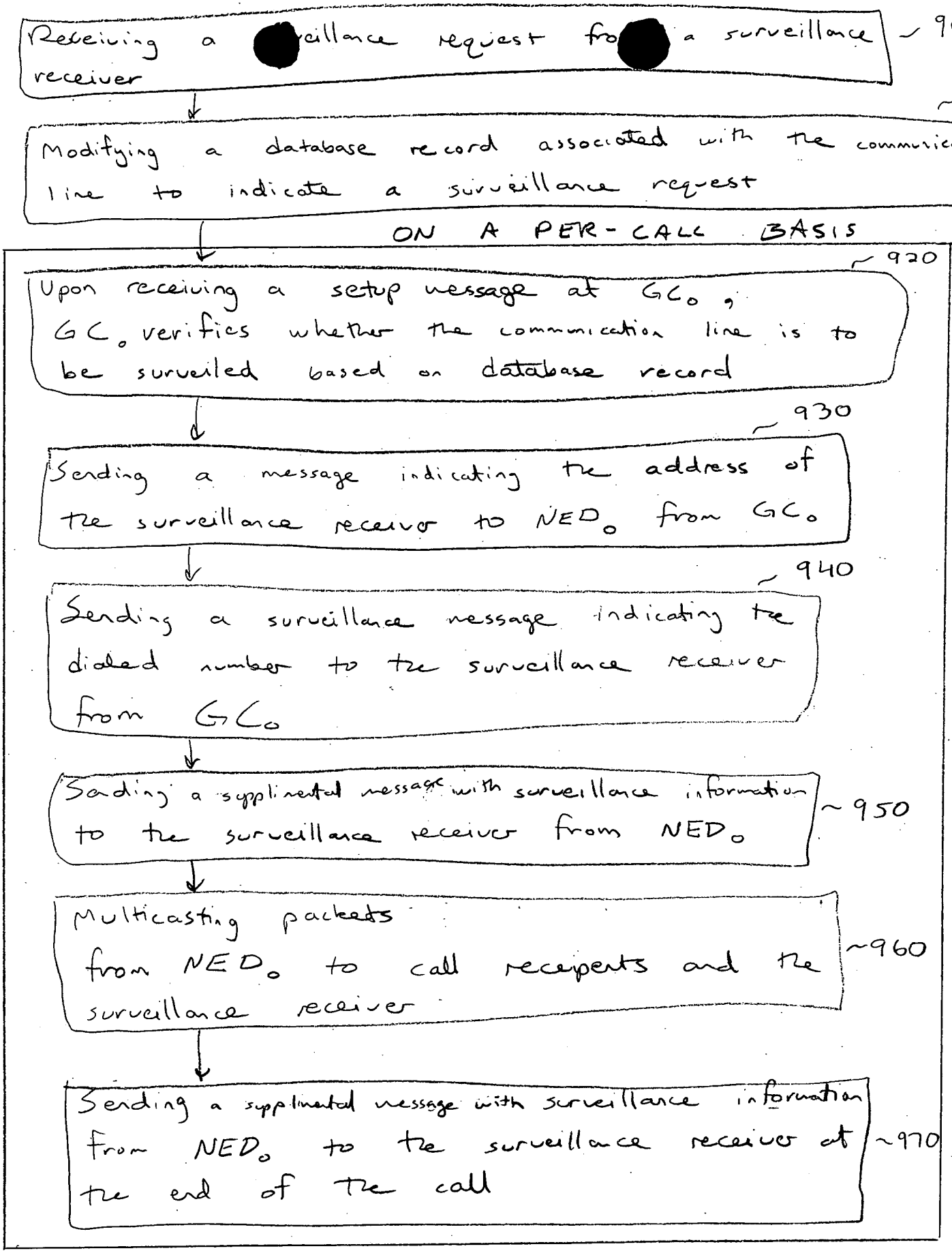
Receiving a surveillance request from a surveillance receiver
900
910
Modifying a database record associated with the communication line to indicate a surveillance request
ON A PER-CALL BASIS
920
Upon receiving a setup message at GC0, GC0 verifies whether the communication line is to be surveiled based on database record
930
Sending a message indicating the address of the surveillance receiver to NED0 from GC0
940
Sending a surveillance message indicating the dialed number to the surveillance receiver from GC0
Sending a supplimental message with surveillance information to the surveillance receiver from NED0
950
Multicasting packets from NED0 to call receipents and the surveillance receiver
960
Sending a supplimental message with surveillance information from NED0 to the surveillance receiver at the end of the call
970

FIG. 36

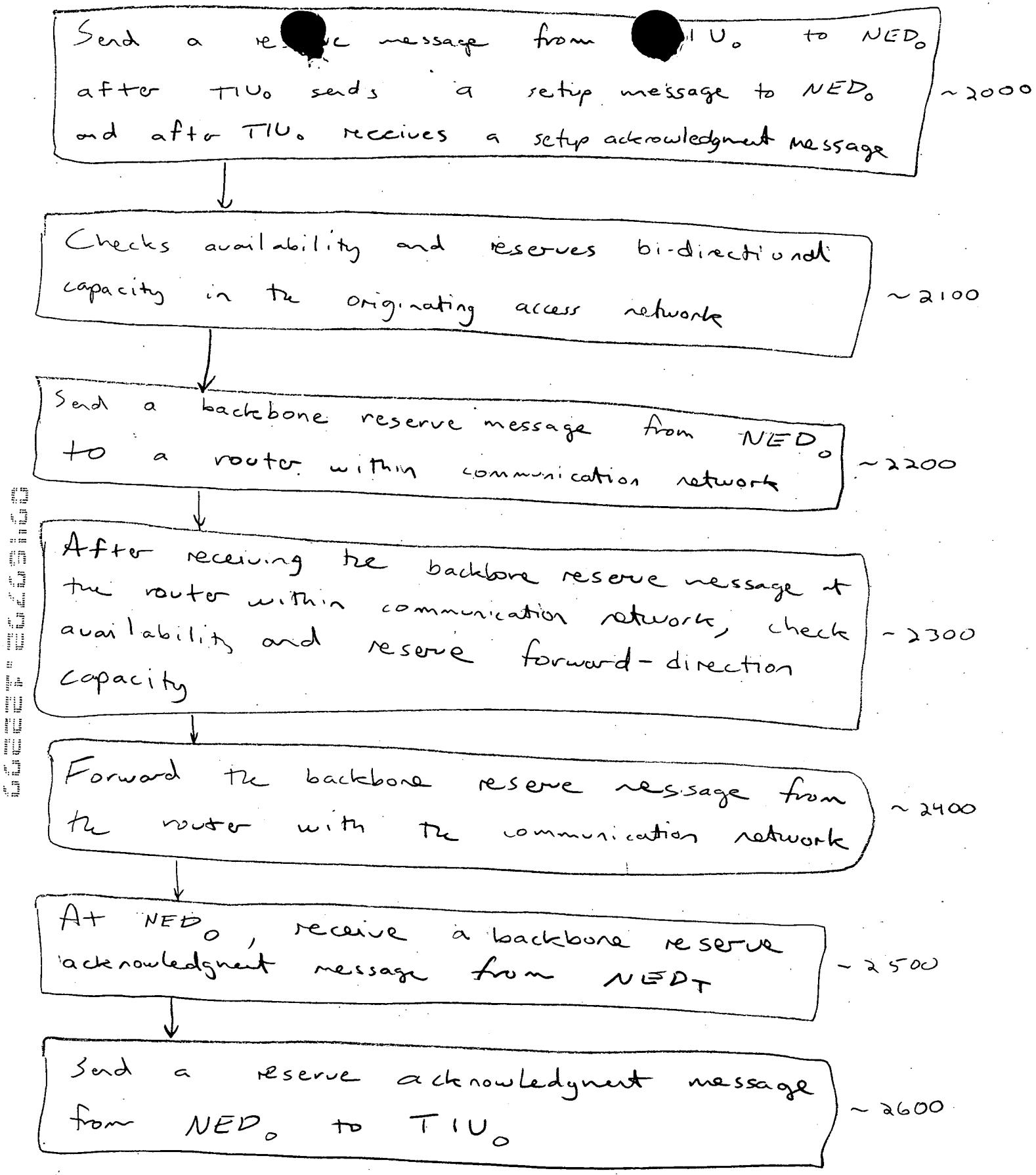
Send a reserve message from TIU$_o$ to NED$_o$ after TIU$_o$ sends a setup message to NED$_o$ and after TIU$_o$ receives a setup acknowledgment message
~2000
Checks availability and reserves bi-directional capacity in the originating access network
~2100
Send a backbone reserve message from NED$_o$ to a router within communication network
~2200
After receiving the backbone reserve message at the router within communication network, check availability and reserve forward-direction capacity
~2300
Forward the backbone reserve message from the router with the communication network
~2400
At NED$_o$, receive a backbone reserve acknowledgment message from NED$_T$
~2500
Send a reserve acknowledgment message from NED$_o$ to TIU$_o$
~2600

FIG. 37